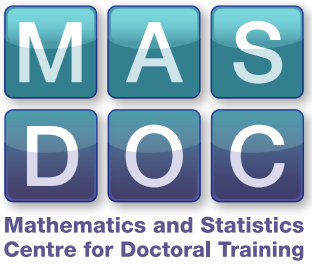


Analysis and Derivation of Continuum $3D$ Blebbing Model with Simulations

by

Adam Nixon

Thesis

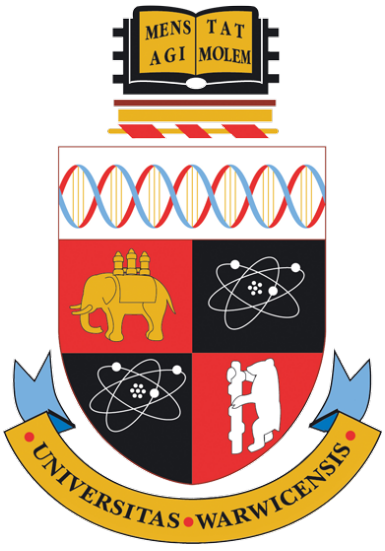
Submitted for the degree of

Doctor of Philosophy

Mathematics Institute

The University of Warwick

September 2018



Contents

List of Tables

List of Figures

Acknowledgments

I am indebted to both of my supervisors Björn Stinner and Andreas Dedner for their guidance, honesty, and at times their considerable patience. I would also like to thank my examiners Anotida Madzvamuse and Marie-Therese Wolfram, who had the perseverance to read, correct and examine me on this thesis.

Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself and has not been submitted in any previous application for any degree.

Work was carried out by myself under the supervision of Björn Stinner with Andreas Dedner providing support for the code development. Additional help was given by Sharon Collier and Till Bretschneider who helped me understand blebs and provide a biologists perspective. Packages and numerical libraries are referenced when appropriate.

Abstract

In this thesis we attempt to give a strong mathematical basis to cellular blebbing models. After outlining the biology and literature surrounding these phenomena we derive a continuum model for a 3D cell. This derivation results in a fourth order parabolic equation with nonlinear second and lower order terms. Using an operator splitting we obtain a system of second order equations. These can be approximated using only linear finite elements. The model generalizes to an abstract formulation.

In this general semi-discrete scheme (continuous in time and discrete in space) we can show weak well posedness and show a priori error estimates. This includes formulating a finite element approximation, proving stability bounds then showing the convergence of the discrete to its limiting case.

We formulate a fully discrete scheme and include a stability proof which allows us to implement the problem in Dune-fem. We can then showcase the implementation by showing the convergence rate of the error on two different surfaces.

Finally, we round off the thesis by showing some more biologically focused examples. Here we use our implementation in two cases and look at the effects of parameters on potential bleb formulations. These use 3D surfaces with no restriction that they be of a certain shape. We also showcase the adaptability of our implementation by applying it to cell image data. We explain how this can be run using Dune-fempy software which offers a lower barrier for entry into creating and editing the scheme. This potentially gives a bridge to test how successful our model or similar models are in modelling blebs.

Chapter 1

Introduction

1.1 Motivation

The membrane of a cell is a very thin layer that defines the boundary of the cell. While the shape of cells is heavily influenced by the cytoskeleton (a network of protein filaments found within most cells), the membrane changes on its own as seen in work observing the shape of vesicles. Understanding the behavior of membranes in cells is a growing area of research which our work can contribute to. We are particularly interested in cell blebbing which helps cells to adapt their migration mechanisms in different environments [21, 30]. Blebs refer to pressure driven spherical protrusions of the membrane. These are small in size and occur over a short time span making them difficult to study experimentally.

Blebbing is part of a cycle (see Figure 1.1) which starts with a membrane that is under pressure and attached to the actin cortex. When the bonds that connect these two are broken, the membrane expands outwards, this is called the expansion stage. Next, during the actin reformation stage, the actin cortex is formed at the membranes new position. A cell may restart the cycle here resulting in a shift of the cell in a certain direction. Otherwise, the cell retracts back to a starting position. Inhibiting this process would be advantageous as it is linked to tumour metastasis [8, 26].

Blebs are more often observed in 3D movement [8] but can be seen on 2D substrates [4, 43]. Movement often seems to be a duality between blebbing and protrusion type locomotion [27, 28]. Some cells use only one form while some use both or switch between them [27]. Blebbing may be linked to exploration of the environment when changing direction, with increased blebbing leading to decreased direction persistence in the migration of zebra-fish development [13]. Recent work

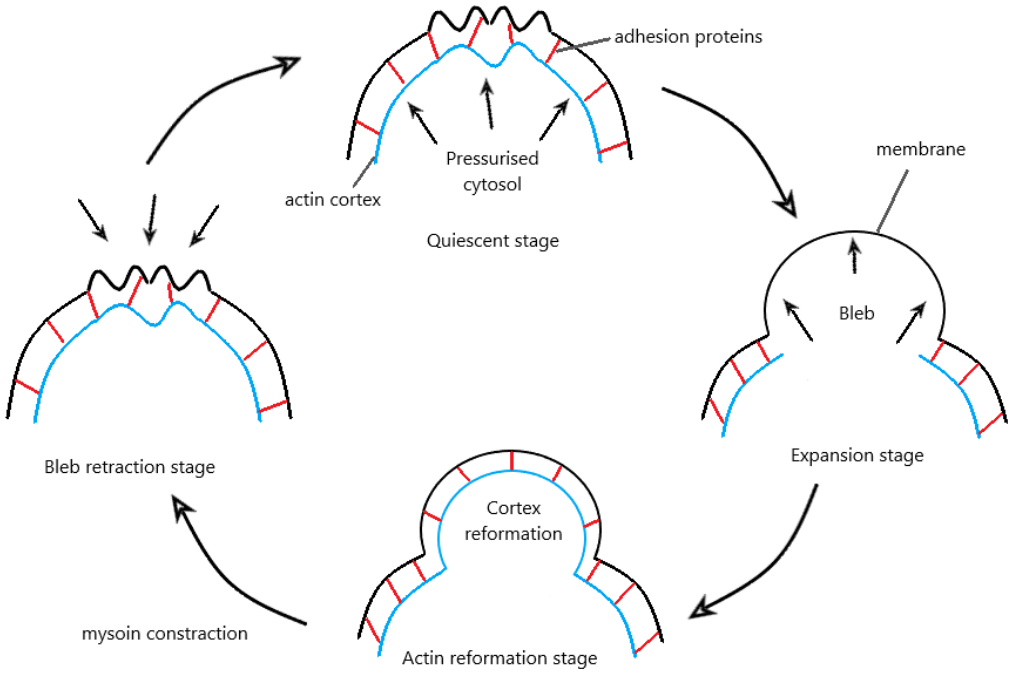


Figure 1.1: When a cell produces a bleb it passes through four phases. First the membrane breaks from the actin cortex and expands outwards. The actin reforms at the new positions and contracts. If the cell is moving by this process no retraction occurs and the cell expands in some new direction. Figure adapted from [8].

in [7] looks to model blebs as a product of the chemical and geometric properties of the cell and whose sites can, therefore, be predicted.

Mechanically before a bleb forms, the cells membrane adheres to a porous mesh known as the cortex, made from polymerized actin chains [8]. This membrane surrounds the intracellular fluid known as cytosol, which is able to flow through the porous cortex. Myosin motors pull on the adhesions connecting the cortex to the membrane causing pressure in the cytosol. If the cortex and/or the adhesions are disrupted, the pressurized cytosol will flow into the weakened section producing a protrusion [41]. The mechanisms that cause the disruption and hence produce a bleb are not known [27, 28]. This work will focus exclusively on the first stage of the bleb cell cycle, so no reforming of the cortex is included. A model for multiple life cycle steps would include the evolution of the actin cortex.

The literature makes several claims that models should conform to, such that in membranes with low tension, blebs cannot be induced [36]. This is difficult to determine experimentally as the link between cortex tension and bleb growth have never been directly assessed [36]. It is also predicted that the cellular membrane

will tear after only a 4% increase in surface area [23]. Finally, there is a prediction that the bending stiffness does not play an important role [41] and that pressure is not a limiting factor in protrusion extension [42]. It is known that whether the membrane protrusion grows or shrinks is determined by cytoplasmic pressure, membrane tension and adhesion energy between the membrane and the cortex [8]. Given these descriptions it is then the purpose of models to be consistent with such theory. Our focus is on providing an adaptable numerical framework for a generalised model than can then be used to compared existing models.

1.2 Overview of Existing Models

Different authors are using various approaches to try modelling cell blebbing. Here is a short overview of both continuous and discrete models. Geometric models are normally 2D with a formulation at the discrete level. Other models focus on small arcs of the membrane and attempt to make analytical conclusions about the forces that drive cell blebs. We give particular focus to the model given in [7] as it will form the basis of work in this thesis.

The first model we shall discuss is the continuum based model presented in [42]. Here, the model is exclusively focused on the surface. The thin membrane is modelled as an elastic sheet. A system of ODEs is used to characterise the force balance of the membrane. They do not consider past the expansion stage. This model often gives large increases in volume, so the tension was adapted to allow for greater stretching to prevent this. They show that although blebs are driven by a pressure difference across the cellular membrane, it is not the limiting factor in bleb size. Also, parameters such as the bending had little affect on the cell's movement.

In [1] a 2D model is proposed and analysed for membrane peeling. The simple model aims to understand the nucleation of blebs by controlling the linker kinetics determining growth and decay when the surface becomes detached. Here an small flat pressurised segment of the membrane is bound to a static cortex by a density of linkers. These bounds are elastic with detachment and reattachment defined by a dynamical system of equations. This system is then analysed for key phase transitions in the pressure and linker density. The critical pressure size is found to depend only on the kinetics of the linkers and the force they withstand. Above this point the detachment grows into a larger bleb and below the separation heals itself.

Using a 2D spherical model and experimental data [36] makes several conclusions about the role of tension. By using lasers to artificially cut the linkers

connecting the cortex and membrane they could induce bleb formation at different tensions. They found if the tension is below a certain threshold blebs do not expand. The paper also concludes that the size of a cellular bleb results from the net effect of cortical tension. This drives the expansion of the bleb, and of its cellular elasticity and membrane tensions, which resist expansion. A simple spherical elastic model is used to relate these experimental results to the cellular mechanisms that drive the formulation.

In [34] a computational model includes changes in the cortex and simulates the whole of a bleb cycle. Here the cortex is an elastic body that is attached by discrete springs to the membrane. These springs do not break. The deformations of the model are defined as a result of the cytoplasmic rheology i.e the internal flow of fluid inside the membrane. The cytosol is assumed to be a viscous fluid modelled by Stokes flow, which is driven by membrane forces. A separate force balance equation describes the movement of the cortex. The improved accuracy in the bulk comes at the cost of computational speed. While multi-blebbing can occur, these are modelled inaccurately. Secondary daughter blebs should be much smaller [36] which does not happen in [34].

Another model that uses Stokes flow is [35] which uses 2D curves for defining the membrane and cortex. Here the strength of the coupling depends on a uniform random distribution. It will be this random factor that decides the bleb's location. Once the spring stretched too far it breaks and its coefficient set to zero. Reformation of the actin cortex is also included by moving the curve representing the cortex closer to the membrane. Once sufficiently close the linker is considered reformed. The forces on the curves are only defined for discrete points with no non discrete formulation given.

Another 2D computational model is given in [44]. Here retraction is included. The membrane itself is modelled by the dampened wave equation with forces from fluid pressure, viscosity, and the filaments added as source terms. The filament forces are computed from Hooke's law. A Lagrangian multiplier is added to kept the volume approximately constant. The model includes breaking of linker by removing a small number of adjacent filaments. At each time step a percentage of filaments are reformed. This reformation then gives a inward pointing force that retracts the bleb.

1.2.1 Finite Difference Scheme

In [7, 38] a 2D discrete model is proposed for locating cell blebs in Dictyostelium cells. This is a discrete model, only ever formulating the membrane as a closed

series of springs which follows a force balance. Therein Dictyostelium is used as the model organism as blebbing can be induced by changing its environment [31] and is dependent on its geometry for movement [7]. Here we outline the model and its limitations. Terms will be discussed in more detail in the next chapter during our model derivation.

The model (1.1) consists of elastic terms accounting for the bending and tension in addition to terms for the coupling and the pressure. The coupling refers to the binding of N discrete linkers to the stationary cortex. As the model aim is to predict bleb sites, using a stationary surface is reasonable for Dictyostelium as bleb nucleation and expansion requires less than 0.1 seconds which is much less than cortex remodelling [45]. Using finite different notation (see Definition A.0.1) the model is written in terms of N cyclic nodes at coordinates $\mathbf{x} = (x^1, x^2)$ over a reference interval of length N consisting of equidistant points s_i for $i = 1, .., N$ distance Δs apart. For given positive parameters $\omega, p, l_0, u_B, k_l, k_L, u_{com}, k_b, k_\psi$ we have

$$\begin{aligned}\omega \frac{d}{dt} x^j(s_i) = &- k_l(|\mathbf{x}(s_i) - \mathbf{x}_c(s_i)| - l_0) \frac{(x^j(s_i) - x_c^j(s_i))}{|\mathbf{x}(s_i) - \mathbf{x}_c(s_i)|} \\ &- k_b \partial_{s^4} x^j(s_i) - \frac{p}{A} \nu^j(s_i) + k_\psi \partial_{s^2} x^j(s_i) \\ &- x_0 \frac{k_\psi}{\Delta s^2} \left(\frac{x^j(s_{i+1}) - x^j(s_i)}{|\mathbf{x}(s_{i+1}) - \mathbf{x}(s_i)|} - \frac{x^j(s_i) - x^j(s_{i-1})}{|\mathbf{x}(s_i) - \mathbf{x}(s_{i-1})|} \right). \end{aligned} \tag{1.1}$$

Here A defines the area contained within the curve defined by the discrete points $\mathbf{x}$ with outward normal $\boldsymbol{\nu} = (\nu^1, \nu^2)$, this defines a outward pushing pressure. As the model is formulated in a completely discrete framework, a coupling term is added with positive coefficient k_l where the bonds connecting to the cortex are a finite set of springs that break beyond a certain point. Tension follows a similar definition with each spring connecting nodes along the discrete membrane with resting length x_0. Also a bending force is included with coefficient k_b that acts as a smoothing. Finally the motion of the membrane is modelled as a result of the viscosity ω. In the current formulation, a transition to higher dimensions is impossible due to some of these modelling choices, such as the tension. The main goal is to correctly predict bleb sites [7] not the long-term evolution of the membrane.

We will derive a similar model in Chapter 2 but now based on a continuum model that we then discretise using our chosen numerical method which we discuss next. This will require interpretation of the discrete formulation for how such a force would work in 3D. With a continuum formulation rigorous results such as well posedness of the model and accuracy of a numerical scheme can also be shown.

Neither of which are given for the scheme as presented in [7].

1.3 Numerical Methods

The different structures and properties of mathematical models require different numerical approaches. As the model gives a surface partial differential equation, we look at various options for solving a problem numerically over a curved surface such as those outlined in [17]. As the domain is not flat it must first be approximated by a discrete surface upon which the scheme is then defined.

One approach is to formulate the surface implicitly via a level set function [17]. In test cases we can define the level set but if one seeks to apply the model to any given surface the level set function much be generated for each case. One can then use this level set function to define a degenerate partial differential equation whose solution solves the surface equation on all level sets [17]. By introducing a thickness to the surface the membrane is now three dimensional, hence the problem must be solved in 3D as well. A level set approach would have an advantage if the model contained a bulk component such as modelling the internal fluid flow. This is because the bulk triangulation is already required and can be chosen independently of the level set function.

Another option that exist for surfaces given by a level set is to introduce a thickness for the domain. On this thickened surface a bulk version of the model equation is solved, this is called the diffuse interface method. The approach arises from surface problems modelled with phase field methods [17]. The bulk version of the model equation is given by introducing a family of non-negative, differentiable functions with compact support that spreads the solution over a thickened domain. One can hopefully then prove that by taking this interface to zero, one recovers the original problem.

Other than level sets we can also extend the surface PDE in other ways such that the embedding equation agrees with the original PDE on the surface [17]. The closest point approach extends the surface such that the extended solution is constant along the normals to the surface such as in [9]. This means the Cartesian differential operators agree with the surface intrinsic differential operators on the surface. Implementing this approach for any surface requires knowledge of the closest point in order for it to be computable.

Another possibility is to introduce a parametrisation on a reference domain such as a sphere. The PDE is then re-formulated on the new reference domain and solved. In our work, we chose to use surface finite elements where a triangulated

surface approximates the domain. Here points on the surface define a mesh that the finite element spaces and scheme then operates over. This means we do not need to consider the bulk and can instead work with only triangular elements. The decision to avoid for example finite differences arises from wishing to take advantage of the variational approach used in the model derivation. Theory for this approach can be found in [16] for parabolic equations but relevant theorems and definitions will be contained in this thesis for completeness. This types of approach for modelling cell membranes can be seen in [6, 33]. Here these models use geometric surface finite elements so the solution feeds back into the triangulated domain.

This matches with how image data is formatted to give a triangulated surface. Being able to readily import real data is a distinct advantage of the approach. We can use post processed image data for our model. This is done by reading into the finite element software the triangulated surfaces produced by the post processing software QuimP [5]. This program tracks the cell's shape in time and outputs triangulated data.

1.4 Thesis Contributions

First, we derive the model and explain the underlying assumption of each term. We take the continuum model approach, building on the literature for 2D in particular [7, 38]. In [38] a fully discrete 2D finite difference model is given to predict bleb sites of cells. We develop a full mathematical framework, which generalises the 2D approach for arbitrary smooth surfaces in 3D. The forces we look to include are the bending and tension of the membrane in addition to the internal pressure and a force describing the linkers that bind the membrane to the cortex. This results in a fourth order variational problem. We shall study the weak version of this problem in the thesis. Blebs are expected to form in convex regions of the surface when the linker are not cut artificially or by some internal process. We restrict the time scale of the model to the expansion stage of the bleb cycle (see Figure 1.1) but do not make predictions on the size of blebs that form as we are only focused on location.

The numerical simulations are based on a discretisation using surface finite elements. It is derived from the variational formulation of the problem. Another upshot of this method is the availability of convergence results of the discrete surface and functions defined on such a domain [17] which open up the possibility of answering analytical questions about the model.

We can now pose a semi-discrete scheme for the model using this framework. By defining certain continuity and growth assumptions we can generalize all second

order terms of the model equation to a general class of functions. In principle, the model could be altered in the future while retaining certain analytical results. The goal of the section is to prove well-posedness of the weak form of the problem. We provide energy bounds on the discrete level, which ensure that the discrete solution does not blow up in finite time. We can then show that the discrete scheme does, in fact, converge weakly to the true solution. Finally, we prove uniqueness and thus conclude well posedness of the weak problem. Assuming regularity, a priori error estimates can be obtained for the general scheme.

Next we consider the fully discrete (in time and space) system. The problem contains nonlinear terms so a semi-implicit method is used. We discretise the non-linear terms explicitly and the linear ones implicitly, which gives a linear system. We present the chosen scheme for the specific regularised model with a proof of its stability.

The theoretical error estimates are supported by the results of two numerical test examples. Each example considers a different surface. First a sphere then a nonconvex surface. Error types looked at are the L^2 and H^1 spatial errors for a fixed point in time. In practice, we notice better than expected convergence in both the test cases. We believe this is due to the specific examples chosen and discuss reasons for this when analysing the error.

Next we carry out tests to see how well the model simulates blebs. The chosen application is to assess whether our model is consistent with predictions in the literature so we show simulations of a cell undergoing blebbing with potential geometries. This can be compared to what is expected in the literature from 2D models. The two shapes we use are that of a blood cell and a sphere with an external protrusion. We see that over time the surfaces in both examples move away from their cortices in concave regions as found in [7] for 2D. Our examples indicate that behaviour of the membrane ranges from no linker breakage to breakage of all linkers. Between these two cases exist examples where the parameters cause blebs to form that do not expand to the whole surface. A development of our approach is in handling arbitrary surfaces, to emphasize this we show an example using a mesh based on real cell data and show blebs forming at various timesteps. This is advantageous as the continuum model, numerical method, and the software can be generalised for more advanced examples. Finally we define a model based on the 2D discrete formulation of [38] to show that the continuum model we derived is accurate. To do this we reduce the dimension of our model to 2 and compare results with those seen in [38].

1.4.1 Software Developments

This thesis involved a lot of software development to validate the results and provide examples. To conclude this first chapter we outline what is implemented and list the tools used. The code can be found on my GitHub account (https://github.com/AdamNixon/Warwick).

All versions of the numerical scheme use the DUNE-Fem toolbox [12] which is a modular software for PDEs. The main model was originally built from pre-built examples and has been greatly edited during the thesis.

We include some discussion of a new Python interface to Dune called Dune-FemPy which some of the code uses. This allows people with access to cell mesh data to run their own simulations and test models. We outline the Python script in Chapter 4 and explain how users can run their own experiments with different parameter values. This greatly reduces the barrier of entry to such tasks by requiring much less advanced mathematics and numerical knowledge. However if the user wishes, the functionality exists to edit the model equation without explicit reference to the discretization. This is because the discrete parts such as the basis functions, triangulation and systems matrix are implemented separately from the model equations. We go through some of the code in this thesis and outline how lines could be changed to run the desired model example.

All graphical output such as solutions to models are generated using ParaView [3]. To run a comparison to the existing model from Section 1.2.1 we use MATLAB. A script in MATLAB was also created to convert image files of grids to a format accepted by Dune (.dgf).

1.4.2 Structure of Thesis

At the beginning of each chapter, we outline required definitions and theorems used. The derivation of the model, which we consider in this thesis is prescribed in Chapter 2. In Chapter 3 we formulate a general semi-discrete scheme, for which we prove well-posedness and energy bounds. Next in the chapter we then define the fully discrete and show stability in time. In the following Chapter 4, we discuss the numerical scheme and show convergence of the error. We then show examples of our model producing blebs on cell-like structures. The thesis finishes with a conclusion Chapter 5 to highlight results gained and ask what else could be done in the future.

Chapter 2

A Continuum Model for Blebbing

In this chapter we will define what we mean by a surface and outline the precise calculus required. We then use this established theory to derive the blebbing model that is the focus of our thesis. Various forces act on the membrane of a blebbing cell and contribute to its deformation process. Cells often use an internal pressure to extend its leading edge. How it orients in space, however, is an open question. One idea is that the membrane geometry, in particular, the surface tension plays a role [7].

Various contour models for blebbing exist for 2D. Here we will use, in particular, the model given for curves in [7] that was discussed in Section 1.2.1. This was originally posed in [38]. We will formulate a continuum model that is based on the finite difference of [38] but extends to 3D. The model now defined over a hypersurface tracks the movement in time of the membrane. The approach is not limited to simple geometries such as a sphere and could, in theory, handle any smooth initial hypersurface.

After some notation and definitions we give an outline of the forces we wish to include in our model with some motivation to our thinking. We present a force balance and then describe how to model each force taking some inspiration from the literature. We then make some simplifications which are necessary to show the analytical results presented in the following chapters. Finally, we present a weak formulation of the problem with an order splitting to reduce the model from a fourth to second order problem.

2.1 Calculus on Surfaces

First we introduce some terminology that will be used and built on throughout the thesis. We define geometric constructs such as a hypersurface, the normal and necessary theories in functional analysis applied to surfaces. The following theorems are stated for general n but in this thesis we will only use $n = 1, 2$.

Definition 2.1.1. *Hypersurface [17] Let $k \in \mathbb{N} \cup \{\infty\}$. $\Gamma \subset \mathbb{R}^{n+1}$ is called a C^k hypersurface if, for each point $\mathbf{x} \in \Gamma$, there exists an open set $U \subset \mathbb{R}^{n+1}$ containing $\mathbf{x}_0$ and a function $\phi \in C^k(U)$ with the property that $\nabla\phi \neq 0$ on $\Gamma \cap U$ and such that*

$$U \cap \Gamma = \{\mathbf{x} \in U \quad | \ \phi(\mathbf{x})\}. \tag{2.1}$$

The linear space

$$T_{\mathbf{x}}\Gamma = \{\boldsymbol{\tau} \in \mathbb{R}^{n+1} | \ \exists\gamma : (-\epsilon, \epsilon) \to \mathbb{R}^{n+1} \text{ differentiable,}$$
$$\gamma((-\epsilon, \epsilon)) \subset \Gamma, \gamma(0) = \mathbf{x} \text{ and } \gamma'(0) = \boldsymbol{\tau}\}$$

is called the tangent space to Γ at $\mathbf{x} \in \Gamma$. It is easy to show $T_{\mathbf{x}}\Gamma = |\nabla\phi(\mathbf{x})|^{\perp}$, the set of all vectors that are orthogonal to $\nabla\phi(\mathbf{x})$, where ϕ is as in (2.1). In particular, $T_{\mathbf{x}}\Gamma$ is an n-dimensional subspace of $\mathbb{R}^{n+1}$.

Definition 2.1.2. *[17] A vector $\boldsymbol{\nu}_\Gamma : \Gamma \to \mathbb{R}^{n+1}$ is called a unit normal at $\mathbf{x} \in \Gamma$ if $\boldsymbol{\nu}_\Gamma(\mathbf{x}) \perp T_{\mathbf{x}}\Gamma$ and $|\boldsymbol{\nu}_\Gamma(\mathbf{x})| = 1$. By the notation of Definition 2.1.1, we have*

$$\boldsymbol{\nu}_\Gamma(\mathbf{x}) = \pm\frac{\nabla\phi(\mathbf{x})}{|\nabla\phi(\mathbf{x})|}. \tag{2.2}$$

A C^1 hypersurface is called orientable if there exists a continuous vector field $\boldsymbol{\nu}_\Gamma : \Gamma \to \mathbb{R}^{n+1}$, such that $\boldsymbol{\nu}_\Gamma(\mathbf{x})$ is a unit vector for Γ for all $\mathbf{x} \in \Gamma$.

In the following definition we introduce a small extension of the domain. This will prove useful in the next chapter when we must map between points near the surface and those points on it.

Definition 2.1.3. *[17] Let $\Omega \subset \mathbb{R}^{n+1}$ be bounded and open with exterior normal $\boldsymbol{\nu}_\Gamma$ and assume $\Gamma = \partial\Omega$ is C^k hypersurface (for $k \geq 2$). The oriented distance function for Γ is then*

$$d(\mathbf{x}) := \begin{cases} \inf_{\mathbf{y}\in\Gamma} |\mathbf{x} - \mathbf{y}|, & \mathbf{x} \in \mathbb{R}^{n+1}\backslash\bar{\Omega} \\ -\inf_{\mathbf{y}\in\Gamma} |\mathbf{x} - \mathbf{y}|, & \mathbf{x} \in \Omega. \end{cases}$$

Lemma 2.1.1. *[17] We define*

$$U_\delta := \{\mathbf{x} \in \mathbb{R}^{n+1} \ : \ |d(\mathbf{x})| \leq \delta\}. \tag{2.3}$$

Then $d \in C^k(U)$, and for every point $\mathbf{x} \in U_\delta$ there exists a unique point $\mathbf{p}(\mathbf{x}) \in \Gamma$ such that

$$\mathbf{x} = \mathbf{p}(\mathbf{x}) + d(\mathbf{x})\boldsymbol{\nu}_\Gamma(\mathbf{p}(\mathbf{x})). \tag{2.4}$$

Moreover we have that

$$\nabla d(\mathbf{x}) = \boldsymbol{\nu}_\Gamma(\mathbf{p}(\mathbf{x})), \qquad |\nabla d(\mathbf{x})| = 1, \qquad \text{for } \mathbf{x} \in U_\delta. \tag{2.5}$$

We illustrate Lemma 2.1.1 in Figure 2.1 to show how points on the surface relate to those within the small band.

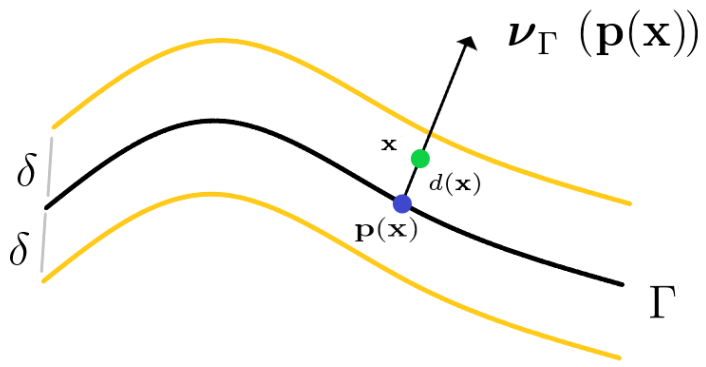


Figure 2.1: By using the distance function d we can extend the surface in the normal direction $\boldsymbol{\nu}_\Gamma$ by a small distance δ.

We can now extend the notations of calculus to fit the setting of a hypersurface.

Definition 2.1.4. *(Surface gradient [15]) For a differentiable function η defined on an open neighbourhood of a C^1 hypersurface Γ we define the surface gradient to be*

$$\nabla_\Gamma \eta(\mathbf{x}) := \nabla\hat{\eta}(\mathbf{x}) - \nabla\hat{\eta}(\mathbf{x}) \cdot \boldsymbol{\nu}_\Gamma(\mathbf{x})\boldsymbol{\nu}_\Gamma(\mathbf{x}) = \mathbf{P}(\mathbf{x})\nabla\hat{\eta}(\mathbf{x}), \qquad \mathbf{x} \in \Gamma,$$

where $\hat{\eta}$ is an extension of η away from Γ and $P(\mathbf{x})_{i,j} = \delta_{i,j} - \nu_{\Gamma,i}(\mathbf{x})\nu_{\Gamma,j}(\mathbf{x})$ for $i,j = 1,..,n+1$. This definition is independent of the choice of extension. We shall also using the following notation

$$\nabla_\Gamma \eta(\mathbf{x}) := \left(\underline{D}_1 f(\mathbf{x}), ..., \underline{D}_{n+1} f(\mathbf{x})\right).$$

For a twice differentiable function we can define the Laplace-Beltrami operator as

$$\Delta_\Gamma \eta(\mathbf{x}) := \nabla_\Gamma \cdot \nabla_\Gamma \eta(\mathbf{x}) = \sum_{i=1}^{n+1} \underline{D}_i \underline{D}_i f(\mathbf{x}).$$

Following on from this we need to define function spaces for which our solution lies.

Definition 2.1.5. *For $p \in [1,\infty]$ and a C^1 hypersurface Γ, let $L^p(\Gamma)$ denote the space of measurable functions $f : \Gamma \to \mathbb{R}$ and have finite norm given by*

$$\begin{aligned} \|f\|_{L^p(\Gamma)} &:= \left(\int_\Gamma |f|^p d\sigma\right) && p < \infty, \\ \|f\|_{L^\infty(\Gamma)} &:= \operatorname*{ess\,sup}_{\mathbf{x}\in\Gamma} |f(\mathbf{x})| && p = \infty. \end{aligned}$$

This naturally leads to the idea of defining Sobolev spaces over the surface. We define the mean curvature H of a C^2 hypersurface Γ as the sum of principle curvatures, $H := \sum_{i=1}^n \kappa_i = \nabla_\Gamma \cdot \boldsymbol{\nu}_\Gamma$.

Definition 2.1.6. *(Weak derivatives [17]) Let Γ by a C^2 hypersurface, then a function $f \in L^1(\Gamma)$ has a weak derivative $\eta_i = \underline{D}_i f \in L^1(\Gamma)$ $(i = \{1,..,n+1\})$ if, for every function $\varphi \in C^1(\Gamma)$ with compact support $\{\mathbf{x} \in \Gamma \mid \varphi(\mathbf{x}) \neq 0\} \subset \Gamma$, we have the following relation*

$$\int_\Gamma f \underline{D}_i \varphi d\sigma = -\int_\Gamma \varphi \eta_i d\sigma + \int_\Gamma f \varphi H \nu_{\Gamma,i} d\sigma.$$

where H is the mean curvature and $\boldsymbol{\nu}_\Gamma = (\nu_{\Gamma,1},..,\nu_{\Gamma,i},..,\nu_{\Gamma,n+1})$ is the outward pointing unit normal.

Definition 2.1.7. *(Sobolev Spaces [2, 22]) For a C^2 hypersurface Γ, we have the Sobolev space*

$$H^1(\Gamma) := \{\eta \in L^2(\Gamma) \mid \nabla_\Gamma \eta \in [L^2(\Gamma)]^{n+1}\}.$$

This is extended, for $k \in \mathbb{N}$ we define

$$H^k(\Gamma) := \{f \in H^{k-1}(\Gamma) \mid \underline{D}_i \eta^{k-1} \in L^2(\Gamma),\ i = 1, .., n-1\}.$$

As our problems are also time dependent so the solution spaces will also be. We, therefore, introduce the time interval $(0, T)$ and work with so-called Bochner spaces.

Definition 2.1.8. *The only values of p we will need are $p = 2, \infty$. Then for a given Sobolev space X*

$$L^p_X := \left\{ f : (0,T) \to X \mid \int_0^T \|f(t)\|^p_X < \infty \right\}.$$

For the case $p = \infty$ we instead use

$$\operatorname*{ess\,sup}_{t \in (0,T)} \|f(t)\|_X < \infty,$$

in the definition.

Lemma 2.1.2. *[10] Let Γ be a C^2 hypersurface with smooth boundary, outer normal $\boldsymbol{\nu}_\Gamma$ and co-normal $\boldsymbol{\mu}_\Gamma$, then for twice differentiable scalar functions $\eta, \xi \in C^2(\bar{\Gamma})$*

$$\int_\Gamma \nabla_\Gamma \eta(\mathbf{x}) \cdot \nabla_\Gamma \xi(\mathbf{x}) d\sigma = \int_{\partial\Gamma} \eta(\mathbf{x}) \nabla_\Gamma \xi(\mathbf{x}) \cdot \boldsymbol{\mu}_\Gamma(\mathbf{x}) d\sigma - \int_\Gamma \xi(\mathbf{x}) \Delta_\Gamma \eta(\mathbf{x}) d\sigma.$$

The solution to our model will define a surface that moves in time. We define an evolving surface in terms of a parametrisation to a fixed reference surface.

Definition 2.1.9. *(Evolving Surface [10]) A family $\{\Gamma\}_{t \in (0,T)}$ is called a $C^{2,1}$-family of hypersurfaces if, for each point $(\mathbf{x}_0, t_0) \in \mathbb{R}^{n+1} \times (0, T)$ with $\mathbf{x}_0 \in \Gamma(t_0)$, there exists an open set $U \subset \mathbb{R}^{n+1}$, $\delta > 0$ and a function $u \in C^{2,1}(U \times (t_0 - \delta, t_0 + \delta))$ such that*

$$U \cap \Gamma(t) = \{\mathbf{x} \in U \mid u(x,t) = 0\} \qquad \text{and} \qquad \nabla u(\mathbf{x}, t) \neq 0,\ \mathbf{x} \in U \cap \Gamma(t).$$

Suppose in addition that each $\Gamma(t)$ is oriented by a unit normal field $\boldsymbol{\nu}_\Gamma(\cdot, t) \in C^1(\Gamma(t), \mathbb{R}^{n+1})$ and that $\boldsymbol{\nu}_\Gamma \in C^0(\cup_{0 \leq t \leq T} \Gamma(t) \times \{t\}, \mathbb{R}^{n+1})$. The normal velocity at a point (x_0, t_0) for $(\mathbf{x}_0 \in \Gamma(t_0)$ is then defined as

$$V(\mathbf{x}_0, t_0) = \phi'(t_0) \boldsymbol{\nu}_\Gamma(\mathbf{x}, t_0),$$

where $\phi \in C^1((t_0 - \epsilon, t_0 + \epsilon), \mathbb{R}^{n+1})$ satisfies $\phi(t) = \mathbf{x}_0$ and $\phi(t_0) \in \Gamma(t)$ for $|t - t_0| \leq \epsilon$.

Further relevant functional analysis theorems can be found in the appendix.

Lemma 2.1.3 ([29], Lemma 7.4)**.** *Let Γ be a C^2 hypersurface and $\boldsymbol{\eta} \in L^2_{H^1}$ with*

$$\int_0^T \int_\Gamma \nabla_\Gamma \boldsymbol{\eta} : \nabla_\Gamma \boldsymbol{\varphi} + \boldsymbol{\eta} \cdot \boldsymbol{\varphi} = 0, \qquad \forall \boldsymbol{\varphi} \in L^2_{H^1},$$

where $:$ denotes the matrix inner product such that $\mathbf{A} : \mathbf{B} = \sum_{i,j=1}^3 a_{i,j} b_{i,j}$ for all $\mathbf{A} : \mathbf{B} \in \mathbb{R}^{3\times 3}$. Then for almost all $t \in (0,T)$

$$\int_\Gamma \nabla_\Gamma \boldsymbol{\eta} : \nabla_\Gamma \boldsymbol{\varphi} + \boldsymbol{\eta} \cdot \boldsymbol{\varphi} = 0, \qquad \forall \boldsymbol{\varphi} \in H^1.$$

To save on notation we will often use the following bilinear forms.

Definition 2.1.10. *Define Γ, a closed C^2 hypersurface without boundary on which we denote the following bilinear forms. For functions $\boldsymbol{\eta}, \boldsymbol{\varphi} \in [L^2(\Gamma)]^3$ define the following*

$$m(\boldsymbol{\eta}, \boldsymbol{\varphi}) := \int_\Gamma \boldsymbol{\eta} \cdot \boldsymbol{\varphi} d\sigma. \tag{2.6}$$

Let $:$ denote the matrix inner product such that $\mathbf{A} : \mathbf{B} = \sum_{i,j=1}^3 a_{i,j} b_{i,j}$ for all $\mathbf{A} : \mathbf{B} \in \mathbb{R}^{3\times 3}$. Then for $\boldsymbol{\eta}, \boldsymbol{\varphi} \in [H^1(\Gamma)]^3$, we also define

$$s(\boldsymbol{\eta}, \boldsymbol{\varphi}) := \int_\Gamma \nabla_\Gamma \boldsymbol{\eta} : \nabla_\Gamma \boldsymbol{\varphi} d\sigma. \tag{2.7}$$

With a slight abuse of notation, for the function $F : \mathbb{R}^{3\times 3} \times \mathbb{R}^{3\times 3} \to \mathbb{R}^{3\times 3}$

$$s(F; \boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \boldsymbol{\varphi}) := \int_\Gamma F(\nabla_\Gamma \boldsymbol{\eta}_1, \nabla_\Gamma \boldsymbol{\eta}_2) : \nabla_\Gamma \boldsymbol{\varphi} d\sigma. \tag{2.8}$$

2.2 Problem Formulation

The blebbing cell occupies an open, simply connected bounded space which changes in time denoted by $\Omega(t)$, where $t \in [0,T]$ with some $T > 0$ stands for time. The evolving boundary $\Gamma(t) := \partial\Omega(t)$, $t \in [0,T]$ describes the position of mass points in the cell membrane. The objective is now to derive a partial differential equation for the membrane. For this purpose, $\Gamma(t)$ is parametrized over the initial surface $\Gamma^0 := \Gamma(0)$, i.e. $\Gamma(t) = \mathbf{u}(\Gamma^0, t)$ for some function $\mathbf{u} : \Gamma^0 \times [0,T] \to \mathbb{R}^3$. The outwards pointing unit normal $\boldsymbol{\nu}_\Gamma$ is chosen for the surface Γ with the special case $\boldsymbol{\nu}_{\Gamma^0}$ for the unit normal at $t = 0$.

2.2.1 The Forces

Considering a force balance to derive bleb models has been used in various examples such as [34, 35]. The terms considered here are the drag due to ambient viscous fluid F_{vis}, bending F_{bend}, tension F_{tens}, pressure F_{pres} and a final term that couples the membranes position to the cortex F_{link}. This yields

$$0 = F_{pres} + F_{vis} + F_{link} + F_{bend} + F_{tens}. \tag{2.9}$$

First, we give a reasoning and explanation for each of these forces. Afterwards, we shall present its mathematical formulation. A full list of parameters introduced with units is stated in Table 2.1.

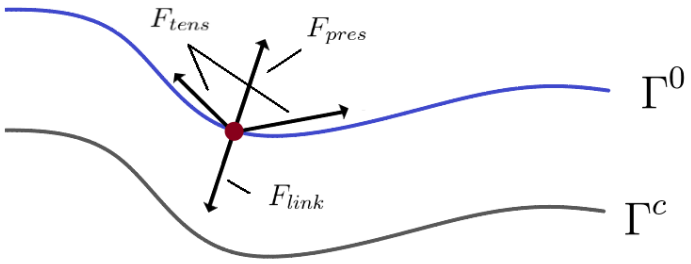


Figure 2.2: We present a sketch of how the forces in (2.9) interact at a given mass point, shown here as a red dot. The pressure is an outwards pushing force that is counteracted by the linkers when attached. This balance is disrupted by the tension force which in concave regions pulls the surface Γ^0 away from the cortex Γ^c. The bending force of omitted here as it is much weaker than these three, if include here it would move the surface in direction of minimising curvature.

The force for pressure reads

$$F_{pres} = \frac{p}{V(\mathbf{u})}\boldsymbol{\nu}_\Gamma, \qquad V(\mathbf{u}) = \frac{1}{3}\int_\Gamma \mathbf{id}\cdot\boldsymbol{\nu}_\Gamma. \tag{2.10}$$

where V is the volume of the cell with a pressure parameter $p > 0$ and the identity map $\mathbf{id} : \mathbb{R}^3 \to \mathbb{R}^3$ is used to compute the volume. The pressure term relates to an internal force pushing against the membrane from inside which decreases as the volume V increases depending on the overall evolution. This force is also of uniform magnitude throughout the cell. Assuming that the deformation $|\mathbf{u} - \mathbf{u}_0|$ is small the

volume can be approximated by the initial size V^0.

$$\begin{aligned} V(\mathbf{u}) =& \frac{1}{3}\int_\Omega \nabla\cdot\mathbf{id} = \frac{1}{3}\int_\Gamma \mathbf{id}\cdot\boldsymbol{\nu}_\Gamma \\ =& \frac{1}{3}\int_{\Gamma^0} \mathbf{u}\cdot\boldsymbol{\nu}_\Gamma\sqrt{\det(\nabla_{\Gamma^0}\mathbf{u}^T\nabla_{\Gamma^0}\mathbf{u})} \\ \approx& \frac{1}{3}\int_{\Gamma^0} \mathbf{u}_0\cdot\boldsymbol{\nu}_{\Gamma^0} = \frac{1}{3}\int_{\Gamma^0}\mathbf{x}\cdot\boldsymbol{\nu}_{\Gamma^0} =: V^0. \end{aligned}$$

The upshot of this formulation is now we require only the normal over the original surface. It has the advantage that it does not degenerate, if the surface looses regularity. By this we mean the function $\mathbf{u}$ could define a non-regular surface. To compute $\boldsymbol{\nu}_{\Gamma^0}$ one can use the cross product of the columns of the matrix $\nabla_{\Gamma^0}\mathbf{u}_0$.

A drag force will oppose the membrane's movement. This viscous force is a measure of resistance when the membrane moves through its environment. We postulate a force of the form

$$F_{visc} = -\omega\partial_t\mathbf{u},$$

where ω relates to the viscosity of the medium's our membrane is moving through. The model is supported by the argument presented in [20] that the viscosity dominates when the thickness is much smaller than the length.

Forces arise from linker molecules connecting the cell membrane with the cell cortex which is a body within the cell that acts as scaffolding to stabilize the cell. In Dictyostelium cells (which are the focus on [7]), bleb nucleation requires less than 0.1 seconds which is much less than cortex remodelling [45]. Hence, we may consider the cortex to be stationary over the time scales that blebs occur as is done in [41]. Linkers serve to hold the membrane in a place close to the cortex. When blebs occur, the membrane moves away from the cortex, and detaches, i.e, the linkers break.

We now add-on a linker term that acts as a coupling between the surface and cortex. A bleb is then the movement of the surface Γ away from the cortex Γ_c. We define Γ_c such that the starting distance between a point on it and Γ^0 is a set distance l_0 similar to [38]. We prescribe a function $\mathbf{u}_c : \Gamma^0 \to \mathbb{R}^3$ such that its image is the cortex $\Gamma_c := \mathbf{u}_c(\Gamma^0)$. This map describes how one point $\mathbf{x} \in \Gamma^0$ connects to a point $\mathbf{u}_c(\mathbf{x})$ on Γ_c. A visualisation of the cortex and membrane can be seen in Figure 2.3 with comparisons to the key linker distances.

We assume that the linkers act like springs with resting length l_0 and that such bonds are dense across the membrane as sizes of an individual bond are small.

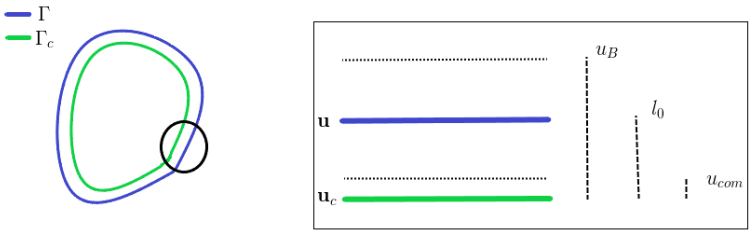


Figure 2.3: The surface defined by $\mathbf{u}$ encapsulates the cortex given by $\mathbf{u}_c$. The linker force changes depending on the distance between the two. Above u_B the linkers are detached, at u_B the two are too close becoming non-physical and finally at l_0 the linkers are assumed to be at resting length.

This allows us to avoid including a finite number of connections between the cortex and membrane. With a static spring constant $k_l > 0$, we thus define a linker energy by

$$E_{link}(\mathbf{u}) := \int_{\Gamma^0} \frac{k_l}{2}(|\mathbf{u} - \mathbf{u}_c| - l_0)^2.$$

A bleb is not formed until these springs are broken, therefore, we so also define a set distance u_B beyond which the linker force vanishes. In regions where all are broken the density of bound linkers is 0. Mathematically this can be done with an indicator function $\mathbf{I}(y)$.

$$\mathbf{I}(y) = \begin{cases} 1, & y \leq 0, \\ 0, & y > 0. \end{cases}$$

We can take $y = |\mathbf{u} - \mathbf{u}_c| - u_B$ to suitably edit the function to fit our model. It is also possible that compression plays a role. If the linker springs compress too much (see Figure 2.3) they lead to an intersection with the cortex. In reality the spring should be much tougher when compressed to a minimum length. At such small values a linear spring model breaks down. A nonlinear linker function could be used, however, we are not aware of any literature that deals with linkers material properties due to their size which would make models difficult to verify. This remains a possible extension of our work however, which is the advantage of allowing more complex breaking behaviour. An alternative solution to preventing intersection with the cortex is to include a dampening that slows down motion towards the cortex beyond a point by increasing ω, this would rule out potential intersection in practise. A final option is to increase the linker strength k_l at small $|\mathbf{u}-\mathbf{u}_c|$ so that it requires

more force to compress the spring.

Here we include a second "jump" once the membrane is sufficiently close to the cortex. This way we do not have to include additional nonlinearities whose affects are unlikely to change the model outcomes. Introduce scalar u_{com} as a modelling parameter that acts at small distances between the cortex and membrane under which any further compression requires much large amounts of force by increasing the linker strength by some large value k_L. The total connection strength is illustrated in Figure 2.4. To reduce notation we write this as the function k such that

$$k(|\mathbf{u}-\mathbf{u}_c|) := (1 + k_L \mathbf{I}(|\mathbf{u}-\mathbf{u}_c| - u_{com}))k_l \mathbf{I}(|\mathbf{u}-\mathbf{u}_c| - u_B). \tag{2.11}$$

The linker force is now defined as the variation of the linker energy as long as the springs are not broken and reads

$$F_{link} := k(|\mathbf{u}-\mathbf{u}_c|)\left(1 - \frac{l_0}{|\mathbf{u}-\mathbf{u}_c|}\right)(\mathbf{u}-\mathbf{u}_c). \tag{2.12}$$

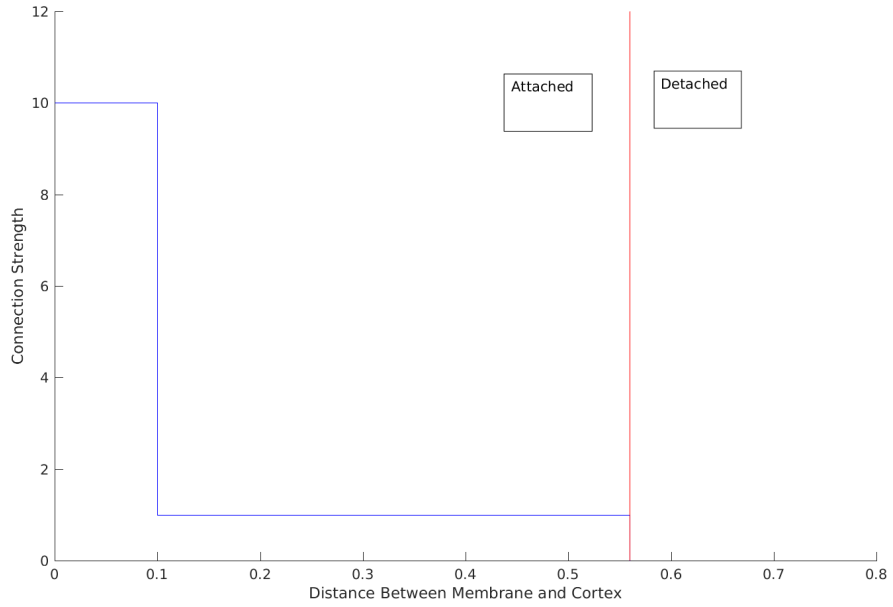


Figure 2.4: A sample plot of the function in (2.11) with parameters $k_L = 9$, $u_{com} = 0.1$, $u_B = 0.56$ and $k_l = 1$. Left of the red line membrane is considered attached and to the right detached.

The idea presented in [7] is that tension causes the location of blebs that are formed as a result of local geometry. Treating the membrane as an elastic body is common since first being introduced in [24] where the energy equation is the mean curvature squared. It is somewhat natural to consider a similar bending force here and include a restriction on tangential stretching such that local area is persevered.

We can now look at the bending and tension terms similar to those included from [25]. As the time scales and deformations are small, it should be reasonable to use a linearised elastic energy for the cell membrane as seen in [38] where a linearisation is also used for the bending. In [41] they suggest the bending acts only as a smoothing. This means that while we include the fourth order term, its effects are assumed to be minor as its coefficient k_b will be small. We define

$$\begin{aligned} E_{elast}(\mathbf{u}) :=& E_{bend}(\mathbf{u}) + E_{tens}(\mathbf{u}) \\ =& \int_{\Gamma^0} \frac{k_b}{2} \left|\Delta_{\Gamma^0}\mathbf{u}\right|^2 + k_\psi \psi(\nabla_{\Gamma^0}\mathbf{u}), \end{aligned} \tag{2.13}$$

where $k_b, k_\psi > 0$ are the bending modulus and the surface tension respectively, and where

$$\psi(\nabla_{\Gamma^0}\mathbf{u}) := \frac{1}{2}(|\nabla_{\Gamma^0}\mathbf{u}| - 2x_0)^2 = \frac{1}{2}(|\nabla_{\Gamma^0}\mathbf{u}| - x_0|\nabla_{\Gamma^0}\mathbf{u}_0|)^2, \tag{2.14}$$

with a given positive scalar $x_0 \in \mathbb{R}$ and the following norm

$$|\mathbf{V}| := \sqrt{\sum_{i,j=1}^{3} v_{i,j} v_{i,j}} \qquad \text{for } \mathbf{V} \in \mathbb{R}^{3\times 3}.$$

The function in (2.14) defines the tension felt at a given point. Here we have used that $\mathbf{u}_0 = \mathbf{x}$ hence $|\nabla_{\Gamma^0}\mathbf{u}_0| = 2$ which means x_0 is relative to the initial area. Similar functions have been used for the area in 2D such as [34] where deviations of the circle are penalised. A spatially dependant function could be used for x_0 but we are not aware of any way to obtain data to calibrate such a function. The forces corresponding to elastic energy are

$$F_{tens} := -\partial E_{tens} \qquad \text{and} \qquad F_{bend} := -\partial E_{bend}.$$

To compute the variation, we look for small deviations $\mathbf{u}_\epsilon := \mathbf{u} + \epsilon\boldsymbol{\phi}$ of $\mathbf{u}$ with $\epsilon << 1$

and $\boldsymbol{\phi} \in C^\infty(\Gamma^0)$. Then

$$\begin{aligned}\langle \partial E_{elast}(\mathbf{u}), \boldsymbol{\phi} \rangle &= \frac{d}{d\epsilon} E_{elast}(\mathbf{u}_\epsilon)\Big|_{\epsilon=0} \\ &= \lim_{\epsilon \to 0} \frac{E_{elast}(\mathbf{u}_\epsilon) - E_{elast}(\mathbf{u})}{\epsilon}.\end{aligned}$$

We have that

$$\begin{aligned}&\langle \partial E_{bend}(\mathbf{u}), \boldsymbol{\phi} \rangle \\ =&\frac{k_b}{2\epsilon} \int_{\Gamma^0} \left[(\Delta_{\Gamma^0}\mathbf{u}) \cdot (\Delta_{\Gamma^0}\mathbf{u}) + 2\epsilon (\Delta_{\Gamma^0}\mathbf{u}) \cdot (\Delta_{\Gamma^0}\boldsymbol{\phi}) + \epsilon^2 (\Delta_{\Gamma^0}\boldsymbol{\phi}) \cdot (\Delta_{\Gamma^0}\boldsymbol{\phi}) \right] ds \Big|_{\epsilon=0} \\ &- \frac{k_b}{2\epsilon} \int_{\Gamma^0} \left[(\Delta_{\Gamma^0}\mathbf{u}) \cdot (\Delta_{\Gamma^0}\mathbf{u}) \right] ds \Big|_{\epsilon=0} \\ =&\frac{k_b}{2\epsilon} \int_{\Gamma^0} \left[2\epsilon (\Delta_{\Gamma^0}\mathbf{u}) \cdot (\Delta_{\Gamma^0}\boldsymbol{\phi}) + \epsilon^2 (\Delta_{\Gamma^0}\boldsymbol{\phi}) \cdot (\Delta_{\Gamma^0}\boldsymbol{\phi}) \right] ds \Big|_{\epsilon=0} \\ =&k_b \int_{\Gamma^0} \left[(\Delta_{\Gamma^0}\mathbf{u}) \cdot (\Delta_{\Gamma^0}\boldsymbol{\phi}) \right] ds \\ =&k_b \int_{\Gamma^0} \Delta^2_{\Gamma^0}\mathbf{u} \cdot \boldsymbol{\phi} ds,\end{aligned}$$

using lemma 2.1.2 twice to get the last line. For the tension term we note that

$$\frac{k_\psi}{2} \int_{\Gamma^0} (|\nabla_{\Gamma^0}\mathbf{u}_\epsilon| - 2x_0)^2 ds = \frac{k_\psi}{2} \int_{\Gamma^0} |\nabla_{\Gamma^0}\mathbf{u}_\epsilon|^2 - 2(2x_0)\,|\nabla_{\Gamma^0}\mathbf{u}_\epsilon| + (2x_0)^2 ds.$$

The first term is the standard Dirichlet energy and the last term is a constant, hence

$$\langle \partial E_{tens}(\mathbf{u}), \boldsymbol{\phi} \rangle = k_\psi \int_{\Gamma^0} \left(\nabla_{\Gamma^0}\mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0}\mathbf{u}}{|\nabla_{\Gamma^0}\mathbf{u}|} \right) : \nabla_{\Gamma^0}\boldsymbol{\phi} ds. \tag{2.15}$$

Altogether, we thus obtain from (2.9) the following PDE problem:

Problem 2.2.1. *(PDE Problem) Let Γ^0 be a given closed hypersurface, then for given constants $\omega, p, l_0, u_B, k_l, k_L, u_{com}, k_b, k_\psi$, and a given function $\mathbf{u}_c : \Gamma^0 \to \mathbb{R}^{n+1}$, find $\mathbf{u} : \Gamma^0 \times [0,T) \to \mathbb{R}^{n+1}$ such that*

$$\begin{aligned}\omega \partial_t \mathbf{u} =& - k_b \Delta^2_{\Gamma^0}\mathbf{u} + k_\psi \nabla_{\Gamma^0} \cdot \left(\nabla_{\Gamma^0}\mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0}\mathbf{u}}{|\nabla_{\Gamma^0}\mathbf{u}|} \right) \\ &- (1 + k_L \mathbf{I}(|\mathbf{u} - \mathbf{u}_c| - u_{com})) k_l \mathbf{I}(|\mathbf{u} - \mathbf{u}_c| - u_B) \left(\mathbf{u} - \mathbf{u}_c - l_0 \frac{\mathbf{u} - \mathbf{u}_c}{|\mathbf{u} - \mathbf{u}_c|} \right) \\ &+ \frac{p}{V^0} \boldsymbol{\nu}_{\Gamma^0},\end{aligned}$$

Parameter	Description	Units
ω	Drag coefficient	$Nm^{-3}s$
x_0	ratio of membrane resting area to starting area	unit less
k_l	Linker strength	Nm^{-3}
k_L	compression parameter	unit less
l_0	Linker resting distance	m
u_B	Braking distance of linkers	m
u_{com}	Minimum cortex-membrane distance	m
k_ψ	Membrane tension stiffness coefficient	Nm^{-1}
k_b	membrane bending modulus	Nm
p	Hydrostatic pressure	Nm

Table 2.1: A full list of all the parameters used in Problem 2.2.1 with units. Some values such are k_L are unit less as they represent an adjustment of k_l which we increase in strength in the event of heavy compression. The ratio x_0 is also dimensionless as it is the ratio of tension relative to the starting configuration $\mathbf{u}_0$.

where

$$V^0 := \frac{1}{3}\int_{\Gamma^0} \mathbf{u}_0 \cdot \boldsymbol{\nu}_{\Gamma^0},$$

and such that $\mathbf{u}(\mathbf{x}, 0) = \mathbf{x}$.

2.3 Non dimensional model equation

As the formulation contains multiple parameters it makes sense to write a non dimensional version. This allows us to reduce the size of the parameter space and let us focus on parameters that we think are most important. The problem we work on is the strong formulation (Problem 2.2.1), which is presented as follows:

$$\omega\partial_t\mathbf{u} - k_\psi\nabla_{\Gamma^0}\cdot\left(\nabla_{\Gamma^0}\mathbf{u} - 2x_0\frac{\nabla_{\Gamma^0}\mathbf{u}}{|\nabla_{\Gamma^0}\mathbf{u}|}\right) + k_b\Delta^2_{\Gamma^0}\mathbf{u}$$
$$+(1 + k_L\mathbf{I}(|\mathbf{u} - \mathbf{u}_c| - u_{com}))k_l\mathbf{I}(|\mathbf{u} - \mathbf{u}_c| - u_B)(\mathbf{u} - \mathbf{u}_c - l_0\frac{\mathbf{u} - \mathbf{u}_c}{|\mathbf{u} - \mathbf{u}_c|}) = \frac{p}{V^0}\boldsymbol{\nu}_{\Gamma^0}.$$

The bending is a smaller parameter than the other terms, so we will consider it of lesser importance in our non dimensionalisation. This means we are focused on how the tension affect the linker and pressure forces. We pick a reference length U and time scale T such that we can define the non dimensional vector and scalar fields $\hat{\mathbf{u}} := \frac{1}{U}\mathbf{u}$, $\hat{\mathbf{x}} := \frac{1}{U}\mathbf{x}$ and $\hat{t} := \frac{1}{T}t$ which allows us to remove the scaling from our

model. We pick U such that it relates to the size of the cell and hence is measured in metres. As we rescale $\mathbf{x}$, this then defines a new scaled surface $\hat{\Gamma}^0$ which has scaled volume, the gradient is now given by $\nabla_{\Gamma^0} = \frac{1}{U}\nabla_{\hat{\Gamma}^0}$ over $\hat{\Gamma}^0$. For the tension and linkers terms we have

$$k_\psi \nabla_{\Gamma^0} \cdot \left(\nabla_{\Gamma^0} \mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0} \mathbf{u}}{|\nabla_{\Gamma^0} \mathbf{u}|} \right) = \frac{k_\psi}{U} \nabla_{\hat{\Gamma}^0} \cdot \left(\nabla_{\hat{\Gamma}^0} \hat{\mathbf{u}} - 2x_0 \frac{\nabla_{\hat{\Gamma}^0} \hat{\mathbf{u}}}{|\nabla_{\hat{\Gamma}^0} \hat{\mathbf{u}}|} \right).$$

Recall that x_0 is a dimensionless parameter, which remains unchanged. Let $\hat{l}_0 := \frac{l_0}{U}$, $\hat{u}_{com} := \frac{u_{com}}{U}$ and $\hat{u}_B := \frac{u_B}{U}$. The indicator function centres around 0 and is invariant under scaling, i.e for some positive scalar $\alpha > 0$, $\mathbf{I}(\alpha x) = \mathbf{I}(x)\ \forall x \in \mathbb{R}$ so does not give a scaling factor in the transformation. For clarity we use $\mathbf{I}(\alpha x) = \hat{\mathbf{I}}(x)$ to indicate that the argument is now non dimensional. The linker term is then

$$\begin{aligned}
&(1 + k_L \mathbf{I}(|\mathbf{u} - \mathbf{u}_c| - u_{com})) k_l \mathbf{I}(|\mathbf{u} - \mathbf{u}_c| - u_B) \left(\mathbf{u} - \mathbf{u}_c - l_0 \frac{\mathbf{u} - \mathbf{u}_c}{|\mathbf{u} - \mathbf{u}_c|} \right) \\
=&(1 + k_L \hat{\mathbf{I}}(|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c| - \hat{u}_{com})) k_l \hat{\mathbf{I}}(|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c| - \hat{u}_B) U \left(\hat{\mathbf{u}} - \hat{\mathbf{u}}_c - \hat{l}_0 \frac{\hat{\mathbf{u}} - \hat{\mathbf{u}}_c}{|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c|} \right).
\end{aligned}$$

We can now present the full equation over the non dimensional surface $\hat{\Gamma}^0$ with outward pointing normal $\boldsymbol{\nu}_{\hat{\Gamma}^0}$:

$$\begin{aligned}
\frac{U\omega}{T} \partial_{\hat{t}} \hat{\mathbf{u}} =& \frac{k_\psi}{U} \nabla_{\hat{\Gamma}_0} \left(\nabla_{\hat{\Gamma}_0} \hat{\mathbf{u}} - 2x_0 \frac{\nabla_{\hat{\Gamma}_0} \hat{\mathbf{u}}}{|\nabla_{\hat{\Gamma}_0} \hat{\mathbf{u}}|} \right) - \frac{k_b}{U^3} \Delta^2_{\hat{\Gamma}_0} \hat{\mathbf{u}} \\
&+ (1 + k_L \hat{\mathbf{I}}(|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c| - \hat{u}_{com})) k_l U \hat{\mathbf{I}}(|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c| - \hat{u}_B) \left(\hat{\mathbf{u}} - \hat{\mathbf{u}}_c - \hat{l}_0 \frac{\hat{\mathbf{u}} - \hat{\mathbf{u}}_c}{|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c|} \right) \\
&+ \frac{p}{U^3 |\hat{V}^0|} \boldsymbol{\nu}_{\hat{\Gamma}^0}.
\end{aligned}$$

This can then be rewritten as

$$\begin{aligned}
\partial_{\hat{t}} \hat{\mathbf{u}} =& \frac{k_\psi T}{U^2 \omega} \nabla_{\hat{\Gamma}_0} \left(\nabla_{\hat{\Gamma}_0} \hat{\mathbf{u}} - 2x_0 \frac{\nabla_{\hat{\Gamma}_0} \hat{\mathbf{u}}}{|\nabla_{\hat{\Gamma}_0} \hat{\mathbf{u}}|} \right) - \frac{k_b T}{U^4 \omega} \Delta^2_{\hat{\Gamma}_0} \hat{\mathbf{u}} \\
&+ \frac{k_l T}{\omega} (1 + k_L \hat{\mathbf{I}}(|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c| - \hat{u}_{com})) \hat{I}(|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c| - \hat{u}_B) \left(\hat{\mathbf{u}} - \hat{\mathbf{u}}_c - \hat{l}_0 \frac{\hat{\mathbf{u}} - \hat{\mathbf{u}}_c}{|\hat{\mathbf{u}} - \hat{\mathbf{u}}_c|} \right) \\
&+ \frac{pT}{U^4 \omega |\hat{V}^0|} \boldsymbol{\nu}_{\hat{\Gamma}^0}.
\end{aligned}$$

Choosing the time scale $T := U^2\omega/k_\psi$ gives

$$\begin{aligned}\partial_{\hat{t}}\hat{\mathbf{u}} =&\nabla_{\hat{\Gamma}_0}\left(\nabla_{\hat{\Gamma}_0}\hat{\mathbf{u}} - 2x_0\frac{\nabla_{\hat{\Gamma}_0}\hat{\mathbf{u}}}{|\nabla_{\hat{\Gamma}_0}\hat{\mathbf{u}}|}\right) - \frac{k_b}{U^2k_\psi}\Delta^2_{\hat{\Gamma}_0}\hat{\mathbf{u}}\\ &+ \frac{k_lU^2}{k_\psi}(1+k_L\hat{\mathbf{I}}(|\hat{\mathbf{u}}-\hat{\mathbf{u}}_c|-\hat{u}_{com}))\hat{\mathbf{I}}(|\hat{\mathbf{u}}-\hat{\mathbf{u}}_c|-\hat{u}_B)\left(\hat{\mathbf{u}}-\hat{\mathbf{u}}_c-\hat{l}_0\frac{\hat{\mathbf{u}}-\hat{\mathbf{u}}_c}{|\hat{\mathbf{u}}-\hat{\mathbf{u}}_c|}\right)\\ &+ \frac{p}{U^2k_\psi|\hat{V}^0|}\boldsymbol{\nu}_{\Gamma^0}.\end{aligned}$$

From here we write each coefficient (for units see Table 2.1) in terms of effective parameters which are non dimensional.

$$\lambda_{b,\psi} := \frac{k_b}{U^2k_\psi}, \qquad \lambda_{l,\psi} := \frac{k_lU^2}{k_\psi}, \qquad \lambda_{p,\psi} := \frac{p}{U^2k_\psi}. \tag{2.16}$$

These dimensionless parameters $\lambda_{l,\psi}, \lambda_{p,\psi}$ relate to the membrane linker attachments. The last parameter, $\lambda_{b,\psi}$ scales with k_b which acts as a smoothing so the overall value will be small. Together this gives the equation

$$\begin{aligned}\partial_{\hat{t}}\hat{\mathbf{u}} =&\nabla_{\hat{\Gamma}_0}\left(\nabla_{\hat{\Gamma}_0}\hat{\mathbf{u}} - 2x_0\frac{\nabla_{\hat{\Gamma}_0}\hat{\mathbf{u}}}{|\nabla_{\hat{\Gamma}_0}\hat{\mathbf{u}}|}\right) - \lambda_{b,\psi}\Delta^2_{\hat{\Gamma}_0}\hat{\mathbf{u}}\\ &+ \lambda_{l,\psi}(1+k_L\hat{\mathbf{I}}(|\hat{\mathbf{u}}-\hat{\mathbf{u}}_c|-\hat{u}_{com}))\hat{\mathbf{I}}(|\hat{\mathbf{u}}-\hat{\mathbf{u}}_c|-\hat{u}_B)\left(\hat{\mathbf{u}}-\hat{\mathbf{u}}_c-\hat{l}_0\frac{\hat{\mathbf{u}}-\hat{\mathbf{u}}_c}{|\hat{\mathbf{u}}-\hat{\mathbf{u}}_c|}\right)\\ &+ \frac{\lambda_{p,\psi}}{|\hat{V}^0|}\boldsymbol{\nu}_{\Gamma^0}.\end{aligned} \tag{2.17}$$

Tweaking parameters such as $\hat{l}_0$ and $\hat{u}_B$ could be done but we avoid discussion here to restrict the parameter space. For the rest of this chapter we drop the hat notation.

2.4 Variational problem formulation

The analysis of PDE Problem 2.2.1 is difficult because of the lack of regularity due to discontinuities that can appear in the pressure, tension, and linker terms. For example our tension term can become discontinuous if $|\nabla_{\Gamma^0}\mathbf{u}| = 0$ as the quotient term becomes not defined.

In order to prove the existence and get error bounds, we consider a regularised model instead. The term for the linkers must also be changed if the membrane $\mathbf{u}$ touches the cortex $\mathbf{u}_c$ i.e $|\mathbf{u}-\mathbf{u}_c| \to 0$. In carrying out any analysis, at least continuity

is required so we decide to regularise some of the functions. This is because our proof will use the convergence of a discrete scheme so we demand certain control of the limit. First we change the tension energy. For $\epsilon_1 > 0$ we approximate the tension energy (from (2.13)) by

$$E_{ten}(\nabla_{\Gamma^0}\mathbf{u}) \approx \frac{1}{2}(\sqrt{\nabla_{\Gamma^0}\mathbf{u} : \nabla_{\Gamma^0}\mathbf{u} + \epsilon_1} - 2x_0)^2,$$

so that

$$F_{tens} \approx \nabla_{\Gamma^0} \cdot \left(\nabla_{\Gamma^0}\mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0}\mathbf{u}}{\sqrt{|\nabla_{\Gamma^0}\mathbf{u}|^2 + \epsilon_1}} \right).$$

With some $\epsilon_2 > 0$ we alter the pressure force (2.10) as follows.

$$F_{pres}(\mathbf{u}) \approx \frac{\lambda_{p,\psi}}{|V^0| + \epsilon_2} \boldsymbol{\nu}_{\Gamma^0}.$$

To avoid issues with regularity of the indicator functions in (2.12) we chose the following approximation using a Heaviside function. For some small $\epsilon_3 > 0$ we approximate (2.11) by

$$k_{\epsilon_3}(y) := \left(1 + \frac{k_L}{1 + e^{\left(\frac{2}{\epsilon_3}(y - u_{com})\right)}}\right) \frac{\lambda_{l,\psi}}{1 + e^{\left(\frac{2}{\epsilon_3}(y - u_B)\right)}}. \tag{2.18}$$

This function is approximately $\lambda_{l,\psi}$ until $y = u_B$ then it quickly transitions to zero. The speed of this change depends on how small ϵ_3 is taken. In Figure 2.4 the function no longer jumps but continuously changes over a small distance of order $(2\epsilon_3)$ between the three values. With this ϵ_2 the connection strength now decreases slightly before the break point is reached. One thus may interpret ϵ_3 as the linkers detaching at a non uniform rate as it reaches the breaking point or a degeneration of the linker bounds as it reaches breaking point. Our approximation of F_{link} (2.12) is now

$$F_{link} \approx k_{\epsilon_3}(|\mathbf{u} - \mathbf{u}_c|) \left(\mathbf{u} - \mathbf{u}_c - l_0 \frac{\mathbf{u} - \mathbf{u}_c}{\sqrt{|\mathbf{u} - \mathbf{u}_c|^2 + \epsilon_3}} \right).$$

Combing all of the discussed terms gives the following problem formulation.

Problem 2.4.1. *For given constants $\lambda_{p,\psi}, l_0, u_B, \lambda_{l,\psi}, k_L, u_{com}, \lambda_{b,\psi}$ and a given Lipschitz function $\mathbf{u}_c : \Gamma^0 \to \mathbb{R}^{n+1}$, find $\mathbf{u} : \Gamma^0 \times [0, T) \to \mathbb{R}^{n+1}$ such that for all test*

functions $\boldsymbol{\varphi}, \boldsymbol{\eta} \in [H^1(\Gamma^0)]^3$ *and all* t

$$\begin{aligned}\int_{\Gamma^0} \frac{\lambda_{p,\psi}}{\epsilon_2 + |V^0|} \boldsymbol{\nu}_{\Gamma^0} \cdot \boldsymbol{\varphi} d\sigma = &\int_{\Gamma^0} \partial_t \mathbf{u} \cdot \boldsymbol{\varphi} + \lambda_{b,\psi} \Delta_{\Gamma^0} \mathbf{u} \cdot \Delta_{\Gamma^0} \boldsymbol{\varphi} d\sigma \\ &+ \int_{\Gamma^0} \left(\nabla_{\Gamma^0} \mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0} \mathbf{u}}{\sqrt{|\nabla_{\Gamma^0} \mathbf{u}|^2 + \epsilon_1}} \right) : \nabla_{\Gamma^0} \boldsymbol{\varphi} d\sigma \\ &+ \int_{\Gamma^0} k_{\epsilon_3}(|\mathbf{u} - \mathbf{u}_c|) \left(\mathbf{u} - \mathbf{u}_c - l_0 \frac{\mathbf{u} - \mathbf{u}_c}{\sqrt{|\mathbf{u} - \mathbf{u}_c|^2 + \epsilon_3}} \right) \cdot \boldsymbol{\varphi} d\sigma,\end{aligned}$$

with $\epsilon_{1,2,3} > 0$*, initial volume denoted as* V^0*,* k_{ϵ_3} *as in (2.18), and such that* $\mathbf{u}(\mathbf{x}, 0) = \mathbf{x} \in \Gamma^0$*.*

2.5 Operator Splitting

To solve Problem 2.4.1 it will be useful to use order reduction to transform the problem into two second-order problems, this would allow us to use linear finite elements. We reduce to second order by introducing a new variable $\mathbf{w}$ which we define as $\mathbf{w} := -\Delta_{\Gamma^0} \mathbf{u}$. Accounting for the regularity we then derive the following weak problem.

Problem 2.5.1. *For given constants* $\lambda_{p,\psi}, l_0, u_B, \lambda_{l,\psi}, k_L, u_{com}, \lambda_{b,\psi}$ *and a given Lipschitz function* $\mathbf{u}_c : \Gamma^0 \to \mathbb{R}^{n+1}$*, find* $(\mathbf{u}, \mathbf{w}) : \Gamma^0 \times [0, T) \to \mathbb{R}^{n+1}$ *such that for all test functions* $\boldsymbol{\varphi}, \boldsymbol{\eta} \in [H^1(\Gamma^0)]^3$ *and all* $t \in [0, T)$

$$\begin{aligned}\int_{\Gamma^0} \frac{\lambda_{p,\psi}}{\epsilon_2 + |V^0|} \boldsymbol{\nu}_{\Gamma^0} \cdot \boldsymbol{\varphi} d\sigma = &\int_{\Gamma^0} \partial_t \mathbf{u} \cdot \boldsymbol{\varphi} + \lambda_{b,\psi} \nabla_{\Gamma^0} \mathbf{w} : \nabla_{\Gamma^0} \boldsymbol{\varphi} d\sigma \\ &+ \int_{\Gamma^0} \left(\nabla_{\Gamma^0} \mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0} \mathbf{u}}{\sqrt{|\nabla_{\Gamma^0} \mathbf{u}|^2 + \epsilon_1}} \right) : \nabla_{\Gamma^0} \boldsymbol{\varphi} d\sigma \\ &+ \int_{\Gamma^0} k_{\epsilon_3}(|\mathbf{u} - \mathbf{u}_c|) \left(\mathbf{u} - \mathbf{u}_c - l_0 \frac{\mathbf{u} - \mathbf{u}_c}{\sqrt{|\mathbf{u} - \mathbf{u}_c|^2 + \epsilon_3}} \right) \cdot \boldsymbol{\varphi} d\sigma, \\ 0 = &\int_{\Gamma^0} \nabla_{\Gamma^0} \mathbf{u} : \nabla_{\Gamma^0} \boldsymbol{\eta} - \mathbf{w} \cdot \boldsymbol{\eta} d\sigma,\end{aligned}$$

with $\epsilon_{1,2,3} > 0$*, initial volume denoted as* V^0*,* k_{ϵ_3} *as in (2.18), and such that* $\mathbf{u}(\mathbf{x}, 0) = \mathbf{x} \in \Gamma^0$*.*

In the following two chapters, we prove weak well posedness for an abstract problem which covers the above. This is first done by showing the existence of a unique solution to a Galerkin approximation of the problem. The results can then be used to show convergence leading to the full Problem 2.5.1.

Defining a new variable $\mathbf{w}$ such that $\mathbf{w} = -\Delta_{\Gamma^0}\mathbf{u}$ holds weakly, allows us to employ continuous piecewise linear finite elements. We also use the finite elements to make an approximation Γ_h^0 of the spatial domain Γ^0.

Chapter 3

Generalised Semi Discrete Approximation

We now look to write an abstract version of Problem 2.5.1. This allows for a well posedness proof that applies to a more general set of equations so that the original function choices could be modelled as desired. The various ϵ dependencies are not tracked through this section as one will gain ϵ^{-1} dependencies, so we have no hope of passing to the limit. The method to prove well posedness is similar to [19] for the Cahn-Hillard model. However the terms of the model differ, in particular for the gradient. After the well posedness proof we obtain error bounds for the generalized semi discrete scheme.

Once we have shown theoretical results for a general problem we must create a fully discrete scheme for our specific case in order to implement the problem and run simulations in future Chapters. We define a semi implicit scheme taking the nonlinear terms explicitly and linear terms implicitly and then give a full matrix vector formulation. We finish the Chapter by obtaining stability bounds for this fully discrete scheme.

3.1 Surface Triangulations and the Lift Operator

To define a problem that is spatially discrete we need a suitable framework. First we must define a surface triangulation such that it approximates the continuum surface as more refinements are made (as $h \to 0$). For more details on the definitions and theorems on surface finite elements see [17].

To set up the domain requires a family of polyhedral surfaces Γ_h^0 that approximates a closed surface Γ^0 as $h \to 0$. Inherent to the approximation will be a

variational crime due to the problem being solved on the surface's approximation. The surface will consist of N vertices, denoted $\mathbf{X}_j \to \mathbb{R}^3$ labelled $j = 1, .., N$. We use the subscript h to indicate discrete quantities. Each vertex is chosen such that it sits on the original surface Γ^0.

Definition 3.1.1. *(Discrete surface [17]) A polyhedral surface is the set generated by the union of finitely many triangles contained within the set* $\mathfrak{T}_h$

$$\Gamma_h^0 = \bigcup_{E \in \mathfrak{T}_h} E \subset \mathbb{R}^{n+1},$$

where E *are (closed) flat* n *dimensional non-degenerate triangles whose pairwise intersection is a complete edge, a single point or empty. For each* $E \in \mathfrak{T}_h$ *denote by* $h(E)$ *its diameter for the grid*

$$h = \max_{E \in \mathfrak{T}_h} h(E).$$

If the discrete surface is to be consistent then we require a suitable projection between Γ_h^0 and Γ^0. By Lemma 2.1.1 we know that there exists a unique curved triangle $e = \mathbf{p}(E) \subset \Gamma^0$. To avoid double covering of Γ^0 we assume this mapping is unique, then for sufficiently small h we can find a unique $\mathbf{p} : \Gamma_h^0 \to \Gamma^0$ such that

$$\mathbf{x} = \mathbf{p}(\mathbf{x}) + d(\mathbf{x})\boldsymbol{\nu}_{\Gamma^0}(\mathbf{p}(\mathbf{x})) \qquad \mathbf{x} \in \Gamma_h^0, \tag{3.1}$$

where d is the oriented distance function to Γ^0 (for illustration of the projection see Figure 3.1).

This also gives for $P(\mathbf{x})_{i,j} = \delta_{i,j} - \nu_i(\mathbf{x})\nu_j(\mathbf{x})$ for $i, j = 1, 2, 3$ the projection matrix onto the tangent space at $\mathbf{x} \in \Gamma^0$

$$\mathbf{P}(\mathbf{x}) = \mathbf{P}(\mathbf{p}(\mathbf{x})).$$

It is sometimes important to give a meaning to a discrete normal $\boldsymbol{\nu}_{\Gamma_h^0}$. The surface Γ_h^0 will always be a surface that is the boundary of a bulk domain. So we can define an outward pointing normal on each face of the discrete surface. This can be estimated in various ways, we use the following definition.

Definition 3.1.2. *We define* $\boldsymbol{\nu}_{\Gamma_h^0}$ *as the outward pointing normal over each discrete face* E *on* Γ_h^0*. In case of a triangle* E *with vertices* $\{\mathbf{q}_i\}_{i=0}^2$ *it is given by*

$$\boldsymbol{\nu}_{\Gamma_h^0}\Big|_E := \frac{(\mathbf{q}_1 - \mathbf{q}_0) \times (\mathbf{q}_2 - \mathbf{q}_0)}{|(\mathbf{q}_1 - \mathbf{q}_0) \times (\mathbf{q}_2 - \mathbf{q}_0)|},$$

where we have assumed that the vertices $\{\mathbf{q}_i\}_{i=0}^2$ *are ordered anti-clockwise on the outer surface of* E*. This is seen visually in Figure 3.2*

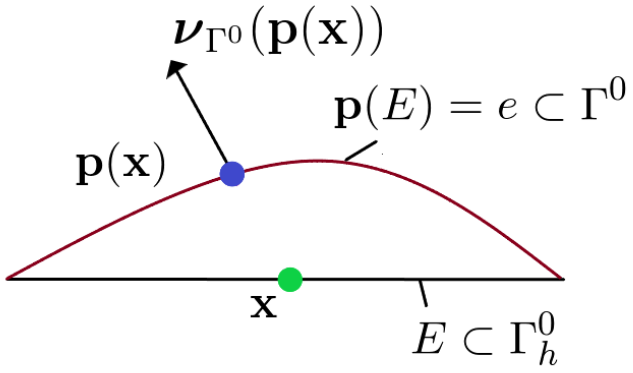


Figure 3.1: The green spot $\mathbf{x} \in \Gamma_h^0$ can be projected onto Γ^0 at the blue spot. This gives a mapping (3.1) between points on the discrete flat triangles E and the curved triangular region it approximates e.

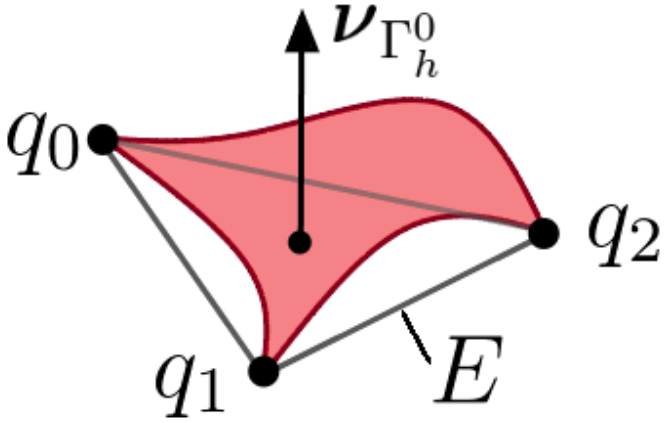


Figure 3.2: Illustration of the unit normal $\boldsymbol{\nu}_{\Gamma_h^0}$ defined in Definition 3.1.2 over a given triangle E of Γ_h^0 by using the vertices $\{\mathbf{q}_i\}_{i=0}^2$ ordered anti-clockwise on the outer surface of E. This normal is constant across each element.

3.1.1 Lift Operator

When working with SFEM we will use a lift operator to switch between working on the discrete domain Γ_h^0 and the original continuous one Γ^0. The following section is found in more detail in [17].

The idea is to create a correspondence between functions on the discrete domain made of triangles $E \in \mathfrak{T}_h$ and the underlining surface Γ^0. By our construction of the triangulation we have that for every $E \subset \Gamma_h^0$ there exists a unique curved triangle $\Gamma^0 \ni e = \mathbf{p}(E) = \{\mathbf{p}(\mathbf{x}) \mid \mathbf{x} \in E)\} \subset \Gamma^0$ (see Figure 3.1). We assume that Γ_h^0 is homeomorphic to Γ^0.

For a function $\eta : \Gamma_h^0 \to \mathbb{R}$, we define its lift $\eta^l : \Gamma^0 \to \mathbb{R}$ by

$$\eta^l(\mathbf{p}(\mathbf{x})) = \eta(\mathbf{x}), \qquad \mathbf{p}(\mathbf{x}) \in \Gamma^0, \ \mathbf{x} \in \Gamma_h^0.$$

Remember that we have the unique map (3.1)

$$\mathbf{x} = \mathbf{p}(\mathbf{x}) + d(\mathbf{x})\boldsymbol{\nu}_{\Gamma^0}(\mathbf{p}(\mathbf{x})), \qquad \mathbf{p}(\mathbf{x}) \in \Gamma^0, \ \mathbf{x} \in \Gamma_h^0.$$

The inverse lift works as follows

$$\eta^{-l}(\mathbf{x}) = \eta(\mathbf{p}(\mathbf{x})), \qquad \mathbf{p}(\mathbf{x}) \in \Gamma^0, \ \mathbf{x} \in \Gamma_h^0.$$

Combing these two returns the original function. This gives us a natural lift of our FE spaces S_h

$$S_h^l := \{\varphi_h = \phi_h^l \,|\phi_h \in S_h\}. \tag{3.2}$$

Lemma 3.1.1. *(Stability of lift [17]) Let $\eta : \Gamma_h^0 \to \mathbb{R}$ with lift $\eta^l : \Gamma^0 \to \mathbb{R}$. Then for an element $E \in \mathfrak{T}_h$, and the related smooth curved triangle $\mathbf{p}(E) = e \subset \Gamma^0$, the following estimates hold if the norms exist. For a constant $c > 0$ independent of h we have*

$$\begin{aligned}
\frac{1}{c}\left\|\eta^l\right\|_{L^2(e)} \leq \|\eta\|_{L^2(E)} \leq c\left\|\eta^l\right\|_{L^2(e)}, \\
\frac{1}{c}\left\|\nabla_{\Gamma^0}\eta^l\right\|_{L^2(e)} \leq \left\|\nabla_{\Gamma_h^0}\eta\right\|_{L^2(E)} \leq c\left\|\nabla_{\Gamma^0}\eta^l\right\|_{L^2(e)}, \\
\left\|\nabla^2_{\Gamma_h^0}\eta\right\|_{L^2(e)} \leq c\left\|\nabla^2_{\Gamma^0}\eta^l\right\|_{L^2(E)} + ch\left\|\nabla^2_{\Gamma^0}\eta^l\right\|_{L^2(E)}.
\end{aligned}$$

Lemma 3.1.1 is very important as it allows us to use bounds derived on Γ_h^0 and the space S_h and create new bounds that are defined over Γ^0 and the finite element space S_h^l.

3.1.2 Geometric estimates

Now we have the definitions for our surface and its discrete approximation it is important to quantify the geometric error. These will be required when proving convergence.

Lemma 3.1.2. *[14, 15, 17] For $\Gamma^0 \in C^2$ and piecewise approximation Γ_h^0, the map $\mathbf{p} : \Gamma_h^0 \to \Gamma^0$ is bijective. For the oriented distance function to Γ^0 we have the estimate*

$$\|d\|_{L^\infty(\Gamma_h^0)} \leq ch^2. \tag{3.3}$$

Let $\delta_h : \Gamma_h^0 \to \mathbb{R}$ denote the ratio of surface measure $d\sigma$ and its discrete counterpart $d\sigma_h$ such that $d\sigma = \delta_h d\sigma_h$. Define the projections $\mathbf{P}$ and $\mathbf{P}_h$ that map onto Γ^0 and Γ_h^0 lets us write the product

$$\mathbf{R}_h := \frac{1}{\delta_h}(\mathbf{I} - d\mathcal{H})\mathbf{P}\mathbf{P}_h\mathbf{P}(\mathbf{I} - d\mathcal{H}),$$

where $\mathcal{H} = \nabla_{\Gamma^0}\boldsymbol{\nu}_{\Gamma^0} = \nabla^2 d$. We then have the following estimates, for some $c > 0$ independent of h

$$\|1 - \delta_h\|_{L^\infty(\Gamma_h^0)} \leq ch^2, \tag{3.4}$$

$$\|(\mathbf{I} - \mathbf{R}_h)\mathbf{P}\|_{L^\infty(\Gamma_h^0)} \leq ch^2, \tag{3.5}$$

$$\left\|\boldsymbol{\nu}_{\Gamma^0} - \boldsymbol{\nu}^l_{\Gamma_h^0}\right\|_{L^\infty(\Gamma^0)} \leq ch, \tag{3.6}$$

$$\left\|\frac{1}{\delta_h}\mathbf{P}_h(\mathbf{I} - d\mathcal{H})\mathbf{P} - \mathbf{P}\right\|_{L^\infty(\Gamma^0)} \leq ch. \tag{3.7}$$

3.2 Surface Finite Elements

Definition 3.2.1. *(Finite Element Space) The standard finite element space used throughout is*

$$S_h := \{\phi \in C^0(\Gamma_h^0) \mid \phi|_E \text{ is linear for each } E \in \mathfrak{T}_h\}.$$

The nodal basis functions of S_h we denote by $\{\phi_i\}_{i=1}^N$ and are characterized by $\phi_i^N(\mathbf{X}_j) = \delta_{i,j}\ \forall i, j = 1, .., N := dim(S_h)$. For vector valued functions the notation will change to S_h^k for a k dimensional vector. These are still linear in each direction of the triangle. For a k dimensional space, we have the indexes $i = 1, .., N_k$ and $\beta = 1, .., k$ for the vector valued function $\boldsymbol{\phi}_{\beta,i} : \Gamma_h^0 \to \mathbb{R}^k$ such that $\boldsymbol{\phi}_{\beta,i} = \phi_i \mathbf{e}_\beta$.

By $\mathbf{e}_\beta$ we denote the β^{th} standard basis vector.

We will also now require a set of bilinear forms similar to Definition 2.1.10 posed over the domain Γ_h^0 with surface measure $d\sigma_h$.

Definition 3.2.2. *For a discrete surface Γ_h^0, let $\boldsymbol{\eta}_h, \boldsymbol{\phi}_h \in [L^2(\Gamma_h^0)]^3$ define the following*

$$m_h(\boldsymbol{\eta}_h, \boldsymbol{\varphi}_h) := \int_{\Gamma_h^0} \boldsymbol{\eta}_h \cdot \boldsymbol{\phi}_h d\sigma_h. \tag{3.8}$$

For $\boldsymbol{\eta}_h, \boldsymbol{\phi}_h \in [S_h(\Gamma_h^0)]^3$ and denoting matrix inner product as :, we also define

$$s_h(\boldsymbol{\eta}_h, \boldsymbol{\phi}_h) := \int_{\Gamma_h^0} \nabla_{\Gamma_h^0} \boldsymbol{\eta}_h : \nabla_{\Gamma_h^0} \boldsymbol{\phi}_h d\sigma_h. \tag{3.9}$$

Using a slight abuse of notation, for the function $F : \mathbb{R}^{3\times 3} \times \mathbb{R}^{3\times 3} \to \mathbb{R}^{3\times 3}$

$$s_h(F; \boldsymbol{\eta}_{1,h}, \boldsymbol{\eta}_{2,h}, \boldsymbol{\varphi}) := \int_{\Gamma_h^0} F(\nabla_{\Gamma_h^0} \boldsymbol{\eta}_{1,h}, \nabla_{\Gamma_h^0} \boldsymbol{\eta}_{2,h}) : \nabla_{\Gamma_h^0} \boldsymbol{\phi}_h d\sigma_h. \tag{3.10}$$

Throughout we will use a Lagrangian interpolation I_h which has the following estimates.

Lemma 3.2.1. *(Interpolation [17]) For $n \leq 3$ and given $\eta \in H^2(\Gamma^0)$, there exists a unique $I_h\eta \in S_h^l$ such that*

$$\|\eta - I_h\eta\|_{L^2(\Gamma^0)} + h \|\nabla_{\Gamma^0} (\eta - I_h\eta)\|_{L^2(\Gamma^0)} \leq ch^2 \left(\left\|\nabla^2_{\Gamma^0}\eta\right\|_{L^2(\Gamma^0)} + h \|\nabla_{\Gamma^0}\eta\|_{L^2(\Gamma^0)} \right).$$

We now also define a discrete projection which is similar to Lagrangian interpolation but allows for less restrictions on the gradient.

Definition 3.2.3 (Ritz Projection [17])**.** *We define for $\eta \in H^1(\Gamma^0)$ the Ritz projection $\Pi_h\eta \in S_h$ such that*

$$\int_0^T s_h(\Pi_h\eta, \phi_h)dt = \int_0^T s(\eta, \phi_h^l)dt \qquad \text{for all } \phi_h \in S_h, \tag{3.11}$$

which is made unique by requiring that

$$\int_{\Gamma_h^0} \Pi_h\eta d\sigma_h d\sigma = \int_{\Gamma^0} \eta d\sigma.$$

Denote the lift of this as

$$\pi_h \eta \equiv (\Pi_h \eta)^l. \tag{3.12}$$

Theorem 3.2.1. *For* $\mathbf{u} \in H^1(\Gamma^0)$,

$$\|\pi_h \mathbf{u}\|_{H^1} \leq c \|\mathbf{u}\|_{H^1}, \qquad \|\pi_h \mathbf{u} - \mathbf{u}\|_{L^2} \leq ch \|\mathbf{u}\|_{H^1},$$

and convergence in the H^1 *norm*

$$\|\pi_h \mathbf{u} - \mathbf{u}\|_{H^1} \to 0, \qquad h \to 0. \tag{3.13}$$

For $\mathbf{u} \in H^2$

$$\|\pi_h \mathbf{u} - \mathbf{u}\|_{L^2} + h \|\nabla_{\Gamma^0}(\pi_h \mathbf{u} - \mathbf{u})\|_{L^2} \leq ch^2 \|\mathbf{u}\|_{H^2}. \tag{3.14}$$

Proof. We prove (3.13) here, the rest of the proof is derived in [17]. For (3.13) we have that H^2 is dense in H^1 so for any $\mathbf{u} \in H^1$ there exists a $\mathbf{v} \in H^2$ such that $\|\mathbf{v} - \mathbf{u}\|_{H^1} \leq \epsilon$ with $\|\mathbf{v} - I_h \mathbf{v}\|_{H^1} \leq Ch$. We can choose h with $Ch \leq \epsilon$. Then with constant $\hat{C}$ from Cea's lemma and denoting $\mathbf{v}_h$ as the interpolation of $\mathbf{v}$ we have

$$\begin{aligned}
\|\mathbf{u} - \pi_h \mathbf{u}\|_{H^1} \leq & \hat{C} \inf_{\mathbf{v}_h \in S_h^l} \|\mathbf{u} - \mathbf{v}_h\|_{H^1} \\
\leq & \hat{C} \|\mathbf{u} - I_h \mathbf{v}\|_{H^1} \\
\leq & \hat{C} \|\mathbf{u} - \mathbf{v}\|_{H^1} + \hat{C} \|\mathbf{v} - I_h \mathbf{v}\|_{H^1} \leq 2\hat{C}\epsilon.
\end{aligned}$$

□

Lemma 3.2.2. *[17] For* $\mathbf{u} : \Gamma^0 \times [0, T) \to \mathbb{R}$ *with* $\mathbf{u}, \partial_t \mathbf{u} \in H^2$ *we have*

$$\|\partial_t(\pi_h \mathbf{u} - \mathbf{u})\|_{L^2} \leq ch^2 \|\partial_t(\pi_h \mathbf{u} - \mathbf{u})\|_{H^2}. \tag{3.15}$$

Corollary 3.2.1. *[17] For* $\mathbf{u} \in L^\infty$ *the Ritz projection is bounded in* L^∞ *and obtain*

$$\|\Pi_h \mathbf{u}\|_{L^\infty} \leq \|\pi_h \mathbf{u}\|_{L^\infty} \leq c \|\Pi_h \mathbf{u}\|_{H^2}.$$

3.2.1 Convergence results

Lemma 3.2.3. *[14] Let* $\mathbf{Z}_h, \boldsymbol{\phi}_h \in S_h$ *with lifts* $\mathbf{z}_h$ $\boldsymbol{\varphi}_h \in S_h^l$ *i.e* $\mathbf{z}_h := \mathbf{Z}_h^l$ *and* $\boldsymbol{\varphi}_h := \boldsymbol{\phi}_h^l$

then the following estimates hold for the bilinear forms

$$|m_h(\mathbf{Z}_h, \boldsymbol{\phi}_h) - m(\mathbf{z}_h, \boldsymbol{\varphi}_h)| \leq ch^2 \|\mathbf{Z}_h\|_{L^2(\Gamma_h^0)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}, \tag{3.16}$$

$$|s_h(\mathbf{Z}_h, \boldsymbol{\phi}_h) - s(\mathbf{z}_h, \boldsymbol{\varphi}_h)| \leq ch^2 \left\|\nabla_{\Gamma_h^0} \mathbf{Z}_h\right\|_{L^2(\Gamma_h^0)} \left\|\nabla_{\Gamma_h^0} \boldsymbol{\phi}_h\right\|_{L^2(\Gamma_h^0)}. \tag{3.17}$$

We now have the theory to quantify the convergence between a discrete function on Γ_h^0 and its lift on Γ^0 using Lemma 3.2.3. This means instead of showing convergence between the a function $y \in H^1(\Gamma^0)$ and a discrete function $Y_h \in S_h(\Gamma_h^0)$ directly, we can break up the problem. Instead comparing Y_h with its lift $y_h \in S_h^l$ using Lemma 3.2.1. Then separately comparing $y_h \in S_h^l(\Gamma^0)$ with $y \in H^1(\Gamma^0)$, so now we are comparing function of the same domain.

3.3 Formulation of General Problem

Problem 2.5.1 can be written in terms of general functions to describe more arbitrary first and second order terms. For the lower order term the domain is expanded to a $\delta > 0$ band around the surface Γ^0 to allow inputs from both Γ_h^0 and Γ^0, the set is denoted as $D_\delta(\Gamma^0)$.

The problem is then of the following more general form. We define $\psi, \mathbf{k}$ that may now differ from those seen in Problem 2.5.1. For the gradient term we use the notation $\psi_A(\mathbf{A}, \mathbf{B}) := \frac{d}{dA}\psi(\mathbf{A}, \mathbf{B})$ to define the derivative with respect to the first argument where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{3\times 3}$. We first define the following point wise assumptions for which we use the following norms

$$|\mathbf{v}| := \sqrt{\sum_{i=1}^{3} v_i v_i} \qquad \text{for } \mathbf{v} \in \mathbb{R}^3,$$

$$|\mathbf{V}| := \sqrt{\sum_{i,j=1}^{3} v_{i,j} v_{i,j}} \qquad \text{for } \mathbf{V} \in \mathbb{R}^{3\times 3}.$$

Assumption 1. *Let $\psi(A, B) : \mathbb{R}^{3\times 3} \times \mathbb{R}^{3\times 3} \to \mathbb{R}$ be a positive continuously differentiable function such that its partial derivative in the first argument $\psi_A : \mathbb{R}^{3\times 3} \times \mathbb{R}^{3\times 3} \to \mathbb{R}^{3\times 3}$ is Lipschitz in both arguments, i.e there exists some $C_\psi > 0$ such that for all $\mathbf{A}_1, \mathbf{B}_1, \mathbf{A}_2, \mathbf{B}_2 \in \mathbb{R}^{3\times 3}$ with induced inner product norms we obtain*

$$|\psi_A(\mathbf{A}_1, \mathbf{B}_1) - \psi_A(\mathbf{A}_2, \mathbf{B}_1)| \leq C_\psi \left(|\mathbf{A}_1 - \mathbf{A}_2|\, |\mathbf{B}_1|\right), \tag{3.18}$$

$$|\psi_A(\mathbf{A}_1, \mathbf{B}_1) - \psi_A(\mathbf{A}_1, \mathbf{B}_2)| \leq C_\psi \left(|\mathbf{B}_1 - \mathbf{B}_2|\, |\mathbf{A}_1|\right). \tag{3.19}$$

ASSUMPTION 2. *Let* $\mathbf{k}, \mathbf{k}_h : \mathbb{R}^3 \times D_\delta(\Gamma^0) \times S^2 \to \mathbb{R}^3$ *be Lipschitz functions such that for all* $\mathbf{u}, \mathbf{v} \in \mathbb{R}^3$, $\mathbf{x}, \mathbf{y} \in D_\delta(\Gamma^0)$ *and* $\boldsymbol{\nu}_{\Gamma^0,1}, \boldsymbol{\nu}_{\Gamma^0,2} \in S^2$ *with induced inner product norms we obtain*

$$|\mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0,1}) - \mathbf{k}(\mathbf{v}, \mathbf{y}, \boldsymbol{\nu}_{\Gamma^0,2})| \leq C \left(|\mathbf{u} - \mathbf{v}| + |\mathbf{x} - \mathbf{y}| + |\boldsymbol{\nu}_{\Gamma^0,1} - \boldsymbol{\nu}_{\Gamma^0,2}| \right), \quad (3.20)$$

$$|\mathbf{k}_h(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0,1}) - \mathbf{k}_h(\mathbf{v}, \mathbf{y}, \boldsymbol{\nu}_{\Gamma^0,2})| \leq C \left(|\mathbf{u} - \mathbf{v}| + |\mathbf{x} - \mathbf{y}| + |\boldsymbol{\nu}_{\Gamma^0,1} - \boldsymbol{\nu}_{\Gamma^0,2}| \right). \quad (3.21)$$

ASSUMPTION 3. *For* $\mathbf{k}, \mathbf{k}_h : \mathbb{R}^3 \times D_\delta(\Gamma^0) \times S^2 \to \mathbb{R}^3$ *there exists* $C > 0$ *independent of* h, *such that* $\mathbf{k}_h$ *converges to* $\mathbf{k}$ *in* $C^0(\mathbb{R}^3, D_\delta(\Gamma^0), S^2)$ *with rate one as* $h \to 0$, *i.e there is some* $C > 0$ *such that for all* $h > 0$, $\mathbf{u} \in \mathbb{R}^3$, $\mathbf{x} \in D_\delta(\Gamma^0)$ *and* $\boldsymbol{\nu}_{\Gamma^0} \in S^2$

$$|\mathbf{k}_h(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}) - \mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0})| \leq Ch(|\mathbf{u}| + 1). \quad (3.22)$$

Remark 3.3.1. *The Lipschitz property in both Assumptions defines a growth bound. There exists some* $\gamma_\psi \geq 0$ *such that*

$$|\psi_A(\mathbf{A}, \mathbf{B})| \leq C_\psi |\mathbf{A}| |\mathbf{B}| + \gamma_\psi, \quad (3.23)$$

holds for all $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{3\times3}$. *Also, there are* $C_k, \gamma_p \geq 0$ *such that*

$$|\mathbf{k}_h(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0})| + |\mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0})| \leq C_k |\mathbf{u}| + \gamma_p, \quad (3.24)$$

holds for all $(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}) \in \mathbb{R}^3 \times D_\delta(\Gamma^0) \times S^2$.

Problem 3.3.1. *Find* $\mathbf{u}(t), \mathbf{w}(t) \in [L^2(0, T; H^1(\Gamma^0))]^3$ *with* $\partial_t \mathbf{u} \in [L^2(0, T; H^1(\Gamma^0))]^3$ *such that for almost all* $t \in (0, T)$

$$m(\partial_t \mathbf{u}, \boldsymbol{\varphi}) + s(\mathbf{w}, \boldsymbol{\varphi}) + s(\psi_A; \mathbf{u}, \mathbf{u}_0, \boldsymbol{\varphi}) + m(\mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}) = 0, \quad (3.25)$$

$$s(\mathbf{u}, \boldsymbol{\chi}) - m(\mathbf{w}, \boldsymbol{\chi}) = 0 \quad (3.26)$$

holds for all $\boldsymbol{\chi}, \boldsymbol{\varphi} \in [H^1(\Gamma^0)]^3$, *and such that* $\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \in [H^2(\Gamma^0)]^3 \ \forall \mathbf{x} \in \Gamma^0$.

The assumption that $\mathbf{u}_0(\mathbf{x}) \in [H^2(\Gamma^0)]^3$ is somewhat restrictive as we intend to take $\mathbf{u}_0 = \mathbf{id}$ so this in fact restricts the domain. To find a weak solution to a fourth order problem however will require H^2 regularity.

The strategy for our proof will be to show existence and uniqueness in the discrete setting, then by showing the convergence of such a scheme to the continuous problem. Note the domain will now be approximated by a discrete surface as described in Definition 3.1.1 and using finite elements outlined in Section 3.2. This semi-discrete problem is given as follows

Problem 3.3.2 (Semi-discrete Problem)**.** *Given ψ with derivative ψ_A and $\mathbf{k}_h$ as described by Assumptions 1-3. For every $t \in (0,T)$ find $\mathbf{U}_h(t), \mathbf{W}_h(t) \in S_h^3$ such that*

$$m_h(\partial_t \mathbf{U}_h, \boldsymbol{\phi}_h) + s_h(\mathbf{W}_h, \boldsymbol{\phi}_h) + s_h(\psi_A; \mathbf{U}_h, \mathbf{U}_h^0, \boldsymbol{\phi}_h) + m_h(\mathbf{k}_h(\mathbf{U}_h, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h) = 0, \tag{3.27}$$

$$s_h(\mathbf{U}_h, \boldsymbol{\chi}_h) - m_h(\mathbf{W}_h, \boldsymbol{\chi}_h) = 0 \tag{3.28}$$

holds for all $\boldsymbol{\chi}_h, \boldsymbol{\phi}_h \in S_h^3$ and such that $\mathbf{U}_h^0 = \Pi_h \mathbf{u}_0$. Here Π_h is as defined in Definition 3.2.3.

3.4 Well Posedness of the Regularised Problem

We start by checking Assumption 1 to show how Problem 2.5.1 satisfies the condition. The first order function is then for $x_0 \geq 0$ and $\epsilon_1 > 0$

$$\begin{aligned}\frac{1}{2}(\sqrt{|\nabla_{\Gamma^0}\mathbf{u}|^2 + \epsilon_1} - 2x_0)^2 &= \frac{1}{2}(\sqrt{|\nabla_{\Gamma^0}\mathbf{u}|^2 + \epsilon_1} - 2x_0|\nabla_{\Gamma^0}\mathbf{u}_0|)^2 \\ &=: \psi(\nabla_{\Gamma^0}\mathbf{u}, \nabla_{\Gamma^0}\mathbf{u}_0).\end{aligned} \tag{3.29}$$

This is a perturbed form of the function that we show the first variation of in (2.15). Note as a function in $\mathbf{u}_0$ it remains constant as $|\nabla_{\Gamma^0}\mathbf{u}_0| = |\nabla_{\Gamma^0}\mathbf{id}| = 2$ where $\mathbf{id}$ is the identity function. The Lipschitz conditions follow from the function ψ_A being differentiable and that derivative being bounded.

$$|\psi_{AA}(A, \nabla_{\Gamma^0}\mathbf{id})| \leq \left(1 + \frac{2x_0}{\sqrt{\epsilon_1}}\right), \qquad \forall A \in \mathbb{R}^{3\times 3}.$$

Note it is the ϵ dependence that will cause the scheme to fail in the limit. As $\epsilon_1 \to 0$ the bound will increase and we recover the discontinuity of Problem 2.5.1 which make the function no longer Lipschitz. For the growth assumption (3.23) we have for any $\mathbf{A} \in \mathbb{R}^{3\times 3}$

$$|\psi_A(\mathbf{A})| := \left|\mathbf{A} - 2x_0 \frac{\mathbf{A}}{\sqrt{|\mathbf{A}|^2 + \epsilon_1}}\right| \leq |\mathbf{A}| + 2x_0.$$

So in our case $C_\psi = 1$ and $\gamma_\psi = 2x_0$.

For the second term have

$$\mathbf{k}(\mathbf{u},\mathbf{x},\boldsymbol{\nu}) =: k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_c(\mathbf{x}), -l_0\frac{\mathbf{u}-\mathbf{u}_c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|^2+\epsilon_3}}\right) - \frac{\lambda_{p,\psi}}{\epsilon_2+|V^0|}\boldsymbol{\nu},$$

with discrete version

$$\mathbf{k}_h(\mathbf{u},\mathbf{x},\boldsymbol{\nu}) =: k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_h^c(\mathbf{x}), -l_0\frac{\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|^2+\epsilon_3}}\right) - \frac{\lambda_{p,\psi}}{\epsilon_2+|V^0|}\boldsymbol{\nu}.$$

Where $\mathbf{u}_h^c$ is the discrete projection of $\mathbf{u}_c$ such that $\mathbf{u}_h^c = \pi_h\mathbf{u}_c$. This means we have $O(h^2)$ convergence by (3.14), hence Assumption 3 holds.

For the Lipschitz condition we show it holds for both parts individually and hence is true for the summation. The first term is broken down further and we note that if two functions are Lipschitz and bounded then the same holds for their product. The following two functions are bounded

$$k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|), \qquad \text{and} \qquad \frac{\mathbf{u}-\mathbf{u}_c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|^2+\epsilon_3}}.$$

So we need only show that the individual parts are Lipschitz.

$$k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c|)(\mathbf{u}-\mathbf{u}_c) = \left(1+\frac{k_L}{1+e^{\left(\frac{2}{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c|-u_{com})\right)}}\right)\frac{\lambda_{l,\psi}(\mathbf{u}-\mathbf{u}_c)}{1+e^{\left(\frac{2}{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c|-u_B)\right)}}.$$

But this is a product of continuously differentiable functions so we can check if the derivative is bounded. Without loss of generality we can assume $\mathbf{u}_c = \mathbf{0}$, $\lambda_{l,\psi} = 1$, $k_L = 0$ and hence obtain

$$\begin{aligned}
\left|\frac{d}{d\mathbf{u}}\left(\frac{\mathbf{u}}{1+e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}}\right)\right| &= \left|\frac{\mathbf{I}}{1+e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}} - \frac{2e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}\mathbf{u}\mathbf{u}^T}{\epsilon_3|\mathbf{u}|\left(1+e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}\right)^2}\right| \\
&\leq 1 + \left|\frac{2|\mathbf{u}|}{\epsilon_3\left(1+e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}\right)}\frac{e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}}{\left(1+e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}\right)}\right| \\
&\leq 1 + \left|\frac{2|\mathbf{u}|}{\epsilon_3\left(1+e^{(\frac{2}{\epsilon_3}(|\mathbf{u}|-u_B))}\right)}\right| \\
&\leq 1 + \frac{2u_B}{\epsilon_3}.
\end{aligned}$$

Now to look at the pressure, we can fix $\mathbf{u}$ and vary only $\boldsymbol{\nu}$. Let $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2 \in S^2$, then

$$\begin{aligned}
&\left|\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_1-\frac{\lambda_{p,\psi}}{\epsilon_2+|V_2^0|}\boldsymbol{\nu}_2\right| \\
\leq&\left|\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_1-\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_2+\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_2-\frac{\lambda_{p,\psi}}{\epsilon_2+|V_2^0|}\boldsymbol{\nu}_2\right| \\
\leq&\left|\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_1-\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_2\right|-\left|\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}\boldsymbol{\nu}_2-\frac{\lambda_{p,\psi}}{\epsilon_2+|V_2^0|}\boldsymbol{\nu}_2\right| \\
\leq&\frac{\lambda_{p,\psi}}{\epsilon_2+|V_1^0|}|\boldsymbol{\nu}_1-\boldsymbol{\nu}_2|+\frac{\lambda_{p,\psi}}{\epsilon_2^2}\left||V_2^0|-|V_1^0|\right| \\
\leq&\frac{\lambda_{p,\psi}}{\epsilon_2}|\boldsymbol{\nu}_1-\boldsymbol{\nu}_2|+\frac{\lambda_{p,\psi}}{3\epsilon_2^2}\left|\int_{\Gamma^0}|\mathbf{u}_0\boldsymbol{\nu}_2|-|\mathbf{u}_0\boldsymbol{\nu}_1|d\sigma\right|.
\end{aligned}$$

Now we apply the reverse triangle inequality to the second term to obtain

$$\begin{aligned}
&\frac{\lambda_{p,\psi}}{\epsilon_2}|\boldsymbol{\nu}_2-\boldsymbol{\nu}_1|+\frac{\lambda_{p,\psi}}{3\epsilon_2^2}\left|\int_{\Gamma^0}|\mathbf{u}_0\boldsymbol{\nu}_2-\mathbf{u}_0\boldsymbol{\nu}_1|d\sigma\right| \\
\leq&\frac{\lambda_{p,\psi}}{\epsilon_2}|\boldsymbol{\nu}_1-\boldsymbol{\nu}_2|+\frac{\lambda_{p,\psi}\|u_0\|}{3\epsilon_2^2}|\boldsymbol{\nu}_2-\boldsymbol{\nu}_1|.
\end{aligned}$$

Since $|k_{\epsilon_3}(\cdot)| \leq C$ as it is the product of bounded function (see (2.18)) we have

$$\begin{aligned}
&|\mathbf{k}_h(\mathbf{u},\mathbf{x},\boldsymbol{\nu})-\mathbf{k}(\mathbf{u},\mathbf{x},\boldsymbol{\nu})| \\
\leq&\Bigg|k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_h^c(\mathbf{x}),-l_0\frac{\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|^2+\epsilon_3}}\right) \\
&-k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_c(\mathbf{x}),-l_0\frac{\mathbf{u}-\mathbf{u}_c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|^2+\epsilon_3}}\right)\Bigg| \\
\leq&\Bigg|k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_h^c(\mathbf{x}),-l_0\frac{\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|^2+\epsilon_3}}\right) \\
&-k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_c(\mathbf{x}),-l_0\frac{\mathbf{u}-\mathbf{u}_c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|^2+\epsilon_3}}\right)\Bigg| \\
&+\Bigg|k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_c(\mathbf{x}),-l_0\frac{\mathbf{u}-\mathbf{u}_c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|^2+\epsilon_3}}\right) \\
&-k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|)\left(\mathbf{u}-\mathbf{u}_c(\mathbf{x}),-l_0\frac{\mathbf{u}-\mathbf{u}_c(\mathbf{x})}{\sqrt{|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|^2+\epsilon_3}}\right)\Bigg| \\
\leq&C_1\left|\mathbf{u}_c(\mathbf{x})-\mathbf{u}_h^c(\mathbf{x})\right|+C_2l_0(|\mathbf{u}|+1)\left|\mathbf{u}_c(\mathbf{x})-\mathbf{u}_h^c(\mathbf{x})\right| \\
&+C_3\left(1+|\mathbf{u}|\right)|k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_c(\mathbf{x})|)-k_{\epsilon_3}(|\mathbf{u}-\mathbf{u}_h^c(\mathbf{x})|)| \\
\leq&Ch^2(|\mathbf{u}|+1).
\end{aligned}$$

This result can be extended constantly to a $D_\delta(\gamma^0)$ region.

For the growth assumption we leverage the Lipschitz property. We find an upper bound of the growth for $\mathbf{u}$ and $\boldsymbol{\nu}_1$, we take $\boldsymbol{\nu}_2$ as a second arbitrary point in S^2 and hence obtain

$$|\mathbf{k}(\mathbf{u},\boldsymbol{\nu}_1) - \mathbf{k}(\mathbf{u}_c,\boldsymbol{\nu}_2)| = |\mathbf{k}(\mathbf{u},\boldsymbol{\nu}_1) - \mathbf{k}(\mathbf{u}_c,\boldsymbol{\nu}_1) + \mathbf{k}(\mathbf{u}_c,\boldsymbol{\nu}_1) - \mathbf{k}(\mathbf{u}_c,\boldsymbol{\nu}_2)|.$$

So this formulation is equivalent to

$$|\mathbf{k}(\mathbf{u},\boldsymbol{\nu}_1) - \mathbf{k}(\mathbf{u}_c,\boldsymbol{\nu}_2)| \leq |\mathbf{u} - \mathbf{u}_c|.$$

Taking reverse triangle inequality gives

$$\begin{aligned} ||\mathbf{k}(\mathbf{u},\boldsymbol{\nu}_1)| - |\mathbf{k}(\mathbf{u}_c,\boldsymbol{\nu}_2)|| &\leq |\mathbf{u} - \mathbf{u}_c| \\ |\mathbf{k}(\mathbf{u},\boldsymbol{\nu}_1)| &\leq |\mathbf{u} - \mathbf{u}_c| + |\mathbf{k}(\mathbf{0},\boldsymbol{\nu}_2)| \\ &\leq |\mathbf{u}| + |\mathbf{u}_c| + \frac{\lambda_{p,\psi}}{\epsilon_2}\,|\boldsymbol{\nu}_2| =: |\mathbf{u}| + \gamma_p. \end{aligned}$$

3.5 Matrix Vector Formulation of Semi Discrete Problem

Now that the semi discrete Problem 3.3.2 has been derived, we give a matrix vector formulation that will be used to show short time existence and also be helpful later in the fully discrete scheme found in Problem 3.10.1 . To do this we test with nodal basis functions in each coordinate (see Definition 3.2.1). The fields are expressed in terms of the basis functions of $\mathbf{S}_h^3$: with degrees of freedom vector $\boldsymbol{\eta}^{\mathbf{u}}, \boldsymbol{\eta}^{\mathbf{w}} : (0,T) \to \mathbb{R}^{3N}$, that is split in the spacial directions $\beta = 1,2,3$, and discrete nodes $i = 1,..,N := dim(S_h)$, together we obtain

$$\mathbf{U}_h(\mathbf{x},t) := \sum_{\beta=1}^{3}\sum_{i=1}^{N} \eta^{\mathbf{u}}_{(\beta-1)N+i}(t)\phi_i(\mathbf{x})\mathbf{e}_\beta, \quad \mathbf{W}_h(\mathbf{x},t) := \sum_{\beta=1}^{3}\sum_{i=1}^{N} \eta^{\mathbf{w}}_{(\beta-1)N+i}(t)\phi_i(\mathbf{x})\mathbf{e}_\beta,$$

for $(\mathbf{x},t) \in \Gamma_h^0 \times (0,T)$ where $\{\mathbf{e}_\beta\}_{\beta=1}^3$ denote the standard basis vectors. We define now the matrices $\mathbf{M}$ and $\mathbf{S}$. And let $\gamma,\beta = 1,2,3$ and $i,j = 1,..,N$ then the

equations of $\mathbf{M}$ and $\mathbf{S}$ are given by

$$M_{(\beta-1)N+i,(\gamma-1)N+j} = \int_{\Gamma_h^0} \delta_{\gamma,\beta}\phi_i(\mathbf{x})\phi_j(\mathbf{x}), \tag{3.30}$$

$$S_{(\beta-1)N+i,(\gamma-1)N+j} = \int_{\Gamma_h^0} \delta_{\gamma,\beta}\nabla_{\Gamma_h^0}\phi_i(\mathbf{x}) \cdot \nabla_{\Gamma_h^0}\phi_j(\mathbf{x}). \tag{3.31}$$

Overall this gives the following structure for a mass matrix

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}^L & 0 & 0 \\ 0 & \mathbf{M}^L & 0 \\ 0 & 0 & \mathbf{M}^L \end{pmatrix} \in \mathbb{R}^{3N\times 3N} \text{ where } \mathbf{M}^L = \left\{ \int_{\Gamma_h^0} \phi_i(\mathbf{x})\phi_j(\mathbf{x}) \right\}_{i,j=1}^{N}, \tag{3.32}$$

and analogously for the stiffness matrix $\mathbf{S}$. To enhance readability we now write $\boldsymbol{\phi}_{\beta,i} = \phi_i \mathbf{e}_\beta$ so that for the nonlinear term we have

$$\begin{aligned} &F(\boldsymbol{\eta}^{\mathbf{u}}(t))_{(\gamma-1)N+j} \\ &= \int_{\Gamma_h^0} \psi_A \left(\sum_{\beta=1}^{3}\sum_{i=1}^{N} \eta^{\mathbf{u}}_{(\beta-1)N+i}(t)\nabla_{\Gamma_h^0}\boldsymbol{\phi}_{\beta,i}, \sum_{\beta=1}^{3}\sum_{i=1}^{N} \eta^{\mathbf{u}}_{(\beta-1)N+i}(0)\nabla_{\Gamma_h^0}\boldsymbol{\phi}_{\beta,i}(\mathbf{x}) \right) \cdot \nabla_{\Gamma_h^0}\boldsymbol{\phi}_{\gamma,j}(\mathbf{x}). \end{aligned}$$

The lower order terms are treated in a similar fashion

$$K(\boldsymbol{\eta}^{\mathbf{u}}(t))_{(\gamma-1)N+j} := \int_{\Gamma_h^0} \mathbf{k} \left(\sum_{\beta=1}^{3}\sum_{i=1}^{N} (\eta^{\mathbf{u}}_{(\beta-1)N+i}(t)\boldsymbol{\phi}_{\beta,i}(\mathbf{x})), \mathbf{x}, \boldsymbol{\nu}_{\Gamma_h^0}(\mathbf{x}) \right) \boldsymbol{\phi}_{\gamma j}(\mathbf{x}).$$

Here $\boldsymbol{\nu}_{\Gamma_h^0}$ is the outward pointing normal as computed by Definition 3.1.2. The following matrix vector formulation corresponds to Problem 3.3.2

$$\begin{aligned} \mathbf{M}\partial_t\boldsymbol{\eta}^{\mathbf{u}} + \mathbf{S}\boldsymbol{\eta}^{\mathbf{w}} + \mathbf{F}(\boldsymbol{\eta}^{\mathbf{u}}) + \mathbf{K}(\boldsymbol{\eta}^{\mathbf{u}}) &= 0, \\ \mathbf{M}\boldsymbol{\eta}^{\mathbf{w}} - \mathbf{S}\boldsymbol{\eta}^{\mathbf{u}} &= 0, \end{aligned}$$

Eliminating $\boldsymbol{\eta}^{\mathbf{w}}$ we get

$$\mathbf{M}\partial_t\boldsymbol{\eta}^{\mathbf{u}} + \mathbf{S}\mathbf{M}^{-1}\mathbf{S}\boldsymbol{\eta}^{\mathbf{u}} + \mathbf{F}(\boldsymbol{\eta}^{\mathbf{u}}) + \mathbf{K}(\boldsymbol{\eta}^{\mathbf{u}}) = 0. \tag{3.33}$$

For this semi-discrete scheme, we can prove it is well posed. We can then use this result in later sections to argue that this solution does, in fact, converge to Problem 3.3.1.

3.6 Well Posedness

Theorem 3.6.1. *Well posedness of FE Scheme 3.3.2. For given $\mathbf{U}_h^0$ and Γ_h^0, there exists a unique solution pair $(\mathbf{U}_h(t), \mathbf{W}_h(t)) \in C^1([0,T], S_h^3) \times C^1([0,T], S_h^3)$ to Problem 3.3.2. In addition we have the following stability bounds*

$$\sup_{t\in(0,T)} \left(\|\mathbf{U}_h(t)\|^2_{H^1(\Gamma_h^0)} \right) + \int_0^T \|\mathbf{W}_h(t)\|^2_{H^1(\Gamma_h^0)}\, dt \leq C(\mathbf{U}_0), \tag{3.34}$$

where $C > 0$ depends on the initial data, geometry, final time and parameters only.

Proof. This proof consists of two steps, namely proving short time existence and then deriving estimates that allow us to prove well posedness on the whole time interval. These two steps are proved in the following subsections.

3.6.1 Short Time Existence

As Problem 3.3.2 is discrete in space and continuous in time it reduces the problem to being an ODE. Treated as one, the Schur factorization in (3.33) gives the following formulations

$$\partial_t \boldsymbol{\eta}^{\mathbf{u}}(t) = \mathbf{M}^{-1} \left(\mathbf{S}\mathbf{M}^{-1}\mathbf{S}\boldsymbol{\eta}^{\mathbf{u}}(t) + \mathbf{F}(\boldsymbol{\eta}^{\mathbf{u}}(t)) + \mathbf{K}\left(\boldsymbol{\eta}^{\mathbf{u}}(t)\right) \right). \tag{3.35}$$

This is now a equation only in time so now we can apply ODE theory to our problem. The right hand side is continuous due to all the terms being linear or assumed to be via Assumptions 1 and 2. ODE theory then gives us for some $T^0 > 0$ at least one solution $\mathbf{U}_h, \mathbf{W}_h \in C^1(\Gamma_h^0, [0, T^0])$. Let $[0, T^M)$ be the maximal time interval of existence for $\mathbf{U}_h$ if their exists no $\mathbf{V}_h$ such that (3.35) holds over $[0, T^M + \delta)$ for any $\delta > 0$ with $\mathbf{V}_h = \mathbf{U}_h$. The idea then is to extend the maximal time to T. The only way for a solution to degenerate is for $\mathbf{U}_h \to \infty$ as $t \to T^c$ for some point in time $T^c > 0$. We can, however, show bounds that prove this does not happen. We will then conclude that in fact $T^0 = T$.

3.6.2 Energy Bounds

Taking $\boldsymbol{\phi}_h = \mathbf{U}_h$ in (3.27) and $\boldsymbol{\chi}_h = \mathbf{W}_h$ in (3.28) and substituting

$$\frac{1}{2}\frac{d}{dt}\|\mathbf{U}_h\|^2_{L^2} + s_h(\psi_A, \mathbf{U}_h, \mathbf{U}_h^0, \mathbf{U}_h) + \|\mathbf{W}_h\|^2_{L^2} + m_h(\mathbf{k}_h(\mathbf{U}_h, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \mathbf{U}_h) = 0.$$

We use the growth assumption (3.24) and move the $\mathbf{k}_h$ term to right hand side to get a bound in terms of $\mathbf{U}_h$ and the data which we denote via an absorbing constant

C:

$$\frac{1}{2}\frac{d}{dt}\left\|\mathbf{U}_h\right\|_{L^2}^2 + s_h(\psi_A;\mathbf{U}_h,\mathbf{U}_h^0,\mathbf{U}_h) + \left\|\mathbf{W}_h\right\|_{L^2}^2 \leq \frac{C}{2}\left(\left\|\mathbf{U}_h\right\|_{L^2}^2 + 1\right).$$

We also use Assumption 1 that the growth (3.23) of ψ_A is bounded, hence

$$\frac{1}{2}\frac{d}{dt}\left\|\mathbf{U}_h\right\|_{L^2}^2 + \left\|\mathbf{W}_h\right\|_{L^2}^2 \leq C(1+\left\|\mathbf{U}_h\right\|_{L^2}^2) + C_\psi \left\|\nabla_{\Gamma_h^0}\mathbf{U}_h^0\right\|_{L^2}\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2}^2.$$

Taking $\boldsymbol{\chi}_h = \mathbf{U}_h$ in (3.28) allows us to bound $\nabla_{\Gamma^0}\mathbf{U}_h$ in terms of the L^2 norms of $\mathbf{U}_h$ and $\mathbf{W}_h$:

$$\begin{aligned} s_h(\mathbf{U}_h,\mathbf{U}_h) = m_h(\mathbf{W}_h,\mathbf{U}_h) &\leq \left\|\mathbf{W}_h\right\|_{L^2}\left\|\mathbf{U}_h\right\|_{L^2} \\ &\leq \frac{1}{2}\left(\frac{1}{\epsilon}\left\|\mathbf{W}_h\right\|_{L^2}^2 + \epsilon\left\|\mathbf{U}_h\right\|_{L^2}^2\right). \end{aligned} \tag{3.36}$$

We can take $\epsilon = C_\psi$ which allows us to move $\mathbf{W}_h$ to the left hand side, to obtain

$$\frac{1}{2}\frac{d}{dt}\left\|\mathbf{U}_h\right\|_{L^2}^2 + \frac{1}{2}\left\|\mathbf{W}_h\right\|_{L^2}^2 \leq C(1+\left\|\mathbf{U}_h\right\|_{L^2}^2). \tag{3.37}$$

We can apply a Gronwall type argument (Lemma A.0.2) to bound $\mathbf{U}_h$ to define in terms of an integral between 0 and T^0. As we now have a bound of $\mathbf{W}_h$ we also get a bound of the gradient for free (see (3.36)).

For the second bound take $\boldsymbol{\phi}_h = \mathbf{W}_h$ in (3.27) and $\boldsymbol{\chi}_h = \partial_t\mathbf{U}_h$ in (3.28) and add the two identities to obtain

$$s_h(\partial_t\mathbf{U}_h,\mathbf{U}_h) + s_h(\psi_A;\mathbf{U}_h,\mathbf{U}_h^0,\mathbf{W}_h) + s_h(\mathbf{W}_h,\mathbf{W}_h) + m_h(\mathbf{k}_h(\mathbf{U}_h,\mathbf{X}_h,\boldsymbol{\nu}_{\Gamma_h^0}),\mathbf{W}_h) = 0.$$

We bound ψ_A and $\mathbf{k}_h$ on the right hand side then separate from $\mathbf{W}_h$ via Young's inequality to get

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2}^2 + \left\|\nabla_{\Gamma_h^0}\mathbf{W}_h\right\|_{L^2}^2 \leq& \left\|\mathbf{k}_h(\mathbf{U}_h,\mathbf{X}_h,\boldsymbol{\nu}_{\Gamma_h^0})\right\|_{L^2}\left\|\mathbf{W}_h\right\|_{L^2} \\ &+ \left\|\psi_A(\mathbf{U}_h,\mathbf{U}_h^0)\right\|_{L^2}\left\|\nabla_{\Gamma_h^0}\mathbf{W}_h\right\|_{L^2} \\ \leq& \frac{1}{2}\left(\left\|\mathbf{k}_h(\mathbf{U}_h,\mathbf{X}_h,\boldsymbol{\nu}_{\Gamma_h^0})\right\|_{L^2}^2 + \left\|\mathbf{W}_h\right\|_{L^2}^2\right) \\ &+ \frac{1}{2}\left(\left\|\nabla_{\Gamma_h^0}\mathbf{W}_h\right\|_{L^2}^2 + \left\|\psi_A(\mathbf{U}_h,\mathbf{U}_h^0)\right\|_{L^2}^2\right). \end{aligned}$$

Then the term $\nabla_{\Gamma_h^0}\mathbf{W}_h$ can be moved to the left. ψ_A we know that satisfies the growth assumption (3.23), and $\mathbf{k}$ condition (3.24) and $\mathbf{W}_h$ is bounded by the pre-

vious estimate (3.37). Hence we obtain

$$\frac{d}{dt}\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2}^2+\left\|\nabla_{\Gamma_h^0}\mathbf{W}_h\right\|_{L^2}^2\leq C\left(\|\mathbf{U}_h\|_{L^2}^2+1+\|\mathbf{W}_h\|_{L^2}^2+\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2}^2\right),$$
$$\frac{d}{dt}\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2}^2+\left\|\nabla_{\Gamma_h^0}\mathbf{W}_h\right\|_{L^2}^2\leq C\left(\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2}^2+\|\mathbf{U}_h\|_{L^2}^2+1\right).$$

We can now apply a Gronwall type argument to write a bound for the gradients in terms of T^0. Due to these bounds we can extend T^0 in fact to any finite time $T>0$. □

3.7 Further Estimates

As we have now proved that the semi discrete scheme in Problem 3.3.2 has a solution. Next we show convergence to the original Problem 3.3.1. In the remainder of this section, we show some improved bounds that allow us to obtain the convergence of the discrete solutions. We can then obtain a second set of bounds defined over Γ^0 in terms of the lifted functions from the first set of bounds. This will give us some control over functions in the continuous domain Γ^0. First, however, we need one more set of bounds. The structure is similar to one seen in [19] which works for the Cahn-Hilliard equation, here we have a more complex gradient term.

Lemma 3.7.1 (Improved Energy bounds)**.** *Problem 3.3.2 satisfies the following stability bound:*

$$\int_0^T\|\partial_t\mathbf{U}_h\|_{L^2}^2\,dt+\sup_{t\in[0,T]}\left(\int_{\Gamma_h^0}\mathbf{W}_h^2\right)+\sup_{t\in[0,T]}\left(\int_{\Gamma_h^0}|\nabla_{\Gamma_h^0}\mathbf{U}_h|^2\right)\leq C(\mathbf{U}_0).\quad(3.38)$$

Proof. We know from Theorem 3.6.1 that $\mathbf{U}_h(t)\in C^1$, so $\boldsymbol{\eta}^{\mathbf{w}}$ defined by (3.35) exists and hence we can take a derivative of (3.28): For all $\boldsymbol{\chi}_h\in S_h^3$

$$s_h(\partial_t\mathbf{U}_h,\boldsymbol{\chi}_h)=m_h(\partial_t\mathbf{W}_h,\boldsymbol{\chi}_h).\quad(3.39)$$

Let $\boldsymbol{\phi}_h=\partial_t\mathbf{U}_h$ in (3.27) and subtract (3.39) with $\boldsymbol{\chi}_h=\mathbf{W}_h$, which yields

$$\begin{aligned}m_h(\partial_t\mathbf{U}_h,\partial_t\mathbf{U}_h)+m_h(\partial_t\mathbf{W}_h,\mathbf{W}_h)+s_h(\psi_A;\mathbf{U}_h,\mathbf{U}_h^0,\partial_t\mathbf{U}_h)&\\+m_h(\mathbf{k}_h(\mathbf{U}_h,\mathbf{X}_h,\boldsymbol{\nu}_{\Gamma_h^0}),\partial_t\mathbf{U}_h)=0.&\quad(3.40)\end{aligned}$$

Note that

$$s_h(\psi_A; \mathbf{U}_h, \mathbf{U}_h^0, \partial_t \mathbf{U}_h) = \frac{d}{dt} \int_{\Gamma_h^0} \psi(\nabla_{\Gamma_h^0} \mathbf{U}_h, \nabla_{\Gamma_h^0} \mathbf{U}_h^0),$$

and the last term of (3.40) can be bounded by (3.24) as follows

$$|m_h(\mathbf{k}_h(\mathbf{U}_h, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \partial_t \mathbf{U}_h)| \leq \frac{1}{2}(C_k \|\mathbf{U}_h\|^2 + \gamma_p) + \frac{1}{2} \|\partial_t \mathbf{U}_h\|^2 .$$

Thus we obtain that

$$\frac{1}{2} m_h(\partial_t \mathbf{U}_h, \partial_t \mathbf{U}_h) + \frac{d}{dt} \int_{\Gamma_h^0} \psi(\nabla_{\Gamma_h^0} \mathbf{U}_h, \nabla_{\Gamma_h^0} \mathbf{U}_h^0) + \frac{1}{2} \frac{d}{dt} m_h(\mathbf{W}_h, \mathbf{W}_h) \leq C(\mathbf{U}_0),$$

and therefore integrating in time gives a bound for $\partial_t \mathbf{U}_h$. We can also apply a Gronwall type argument to obtain a bound for the remaining terms. Combining these two gives

$$\int_0^T \|\partial_t \mathbf{U}_h\|_{L^2}^2 \, dt + \sup_{t \in [0,T]} \left(\int_{\Gamma_h^0} \mathbf{W}_h^2 + \psi(\nabla_{\Gamma_h^0} \mathbf{U}_h, \nabla_{\Gamma_h^0} \mathbf{U}_h^0) \right) \leq C(\mathbf{U}_0). \tag{3.41}$$

By Assumption 1 the function ψ is positive so can be dropped. In (3.36) we bound the gradient by $\mathbf{U}_h$ and $\mathbf{W}_h$. We know by (3.34) and (3.41) both terms are bounded in time via the supremium norm. So we conclude the same must also hold for the gradient.

□

Now we can use the bounds (3.34)(3.38) to estimate $\mathbf{u}_h := \mathbf{U}_h^l$ and $\mathbf{w}_h := \mathbf{W}_h^l$ which are defined over Γ^0. This gives some control of the discrete solution that is separate from the variational crime committed by replacing Γ^0 with Γ_h^0.

Theorem 3.7.1 (Lifted Estimates)**.** *Let* $(\mathbf{U}_h, \mathbf{W}_h)$ *solve Problem 3.3.2 then lifts* $\mathbf{u}_h := \mathbf{U}_h^l$ *and* $\mathbf{w}_h := \mathbf{W}_h^l$ *satisfy the following stability bound*

$$\sup_{t \in (0,T)} \left(\|\mathbf{u}_h(t)\|_{H^1(\Gamma^0)}^2 \right) + \int_0^T \|\mathbf{w}_h(t)\|_{H^1(\Gamma^0)}^2 \, dt \leq C_1(\mathbf{u}_0). \tag{3.42}$$

Assuming $\mathbf{u}_0 \in [H^2(\Gamma^0)]^3$ *then the following is also true:*

$$\int_0^T \|\partial_t \mathbf{u}_h\|_{L^2(\Gamma^0)}^2 \, dt + \sup_{t \in [0,T]} \left(\int_{\Gamma^0} \mathbf{w}_h^2 \right) + \sup_{t \in [0,T]} \left(\int_{\Gamma^0} |\nabla_{\Gamma^0} \mathbf{u}_h|^2 \right) \leq C_2(\mathbf{u}_0) \tag{3.43}$$

where $C_{1,2} > 0$ *depend on the initial data, geometry, final time and parameters only.*

Proof. We can lift bounds (3.34)(3.38) via the stability properties of Lemma 3.1.1. □

3.8 Well posedness of the continuous problem

We are now working primarily on the continuous domain Γ^0. Throughout this section for the given function $\mathbf{u}_0 \in [H^2(\Gamma^0)]^3$ we take $\mathbf{U}_h^0 = \Pi_h \mathbf{u}_0$.

Theorem 3.8.1 (Well posedness of Problem 3.3.1)**.** *Given $\mathbf{u}_0 \in H^2(\Gamma^0)$ there exists a unique weak solution $\mathbf{u}(t), \mathbf{w}(t) \in [L^2(0,T; H^1(\Gamma^0))]^3$ to Problem 3.3.1 which satisfies the bound*

$$\sup_{t\in(0,T)} \|\mathbf{u}\|^2_{H^1(\Gamma^0)} + \int_0^T \|\mathbf{w}\|^2_{H^1(\Gamma^0)}\, dt \leq C(\mathbf{u}_0). \tag{3.44}$$

Proof. The proof is split into three parts, showing existence, the convergence to the limiting problem, and uniqueness. The proof will be based on the a priori estimates derived in the previous subsection.

3.8.1 Existence

From our bounds (3.42),(3.43) we know that $\mathbf{u}_h$, is uniformly bounded in $L^\infty_{H^1(\Gamma^0)}$ with $\partial_t \mathbf{u}_h \in L^2_{L^2(\Gamma^0)}$ and $\mathbf{w}_h \in L^2_{H^1(\Gamma^0)}$. Hence we can find sub-sequences (which we relabel with h) and functions $\hat{\mathbf{u}}, \hat{\mathbf{w}}$ with $\hat{\mathbf{u}} \in L^2_{H^1(\Gamma^0)}$, $\hat{\mathbf{u}}_t \in L^2_{L^2(\Gamma^0)}$ and $\hat{\mathbf{w}} \in L^2_{H^1(\Gamma^0)}$.

$$\mathbf{u}_h \rightharpoonup \hat{\mathbf{u}} \qquad \text{weakly in } L^2_{H^1(\Gamma^0)}, \tag{3.45}$$

$$\partial_t \mathbf{u}_h \rightharpoonup \partial_t \hat{\mathbf{u}} \qquad \text{weakly in } L^2_{L^2(\Gamma^0)}, \tag{3.46}$$

$$\mathbf{w}_h \rightharpoonup \hat{\mathbf{w}} \qquad \text{weakly in } L^2_{H^1(\Gamma^0)}. \tag{3.47}$$

By compactness (see [32]) we have

$$\mathbf{u}_h \to \hat{\mathbf{u}} \qquad \text{strongly in } L^2_{L^2(\Gamma^0)} \text{ and almost everywhere in } \Gamma^0 \times (0,T). \tag{3.48}$$

For strong convergence of the gradient we can leverage the weak convergence of $\mathbf{w} \to \hat{\mathbf{w}}$. Denoting the lift of the test function as $\boldsymbol{\varphi}_h := \boldsymbol{\phi}_h^l$ we first look to show $\hat{\mathbf{u}}$

solves (3.26) as well.

$$\begin{aligned}\int_0^T s(\hat{\mathbf{u}},\boldsymbol{\varphi}_h) - m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h)dt &= \int_0^T s(\hat{\mathbf{u}},\boldsymbol{\varphi}_h) - s_h(\mathbf{U}_h,\boldsymbol{\phi}_h) - m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h) + m_h(\mathbf{W}_h,\boldsymbol{\phi}_h)dt \\ &= \int_0^T s(\hat{\mathbf{u}},\boldsymbol{\varphi}_h) - s(\mathbf{u}_h,\boldsymbol{\varphi}_h) + \underbrace{s(\mathbf{u}_h,\boldsymbol{\varphi}_h) - s_h(\mathbf{U}_h,\boldsymbol{\phi}_h)}_{I}\, dt \\ &\quad - \int_0^T m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h) - m(\mathbf{w}_h,\boldsymbol{\varphi}_h)dt \\ &\quad - \int_0^T \underbrace{m(\mathbf{w}_h,\boldsymbol{\varphi}_h) - m_h(\mathbf{W}_h,\boldsymbol{\phi}_h)}_{II}\, dt.\end{aligned}$$

The labelled terms I and II are the geometric errors made in the discretization. We can now make use of Lemma 3.2.3 and the estimates (3.42) and (3.43) to bound I and II to obtain

$$\begin{aligned}\int_0^T s(\hat{\mathbf{u}},\boldsymbol{\varphi}_h) - m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h)dt \leq \int_0^T & s(\hat{\mathbf{u}}-\mathbf{u}_h,\boldsymbol{\varphi}_h) + m(\mathbf{w}_h-\hat{\mathbf{w}},\boldsymbol{\varphi}_h) \\ &+ ch^2\left(\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h\right\|_{L^2_{L^2(\Gamma_h^0)}}\left\|\nabla_{\Gamma_h^0}\boldsymbol{\phi}_h\right\|_{L^2_{L^2(\Gamma_h^0)}}\right) \\ &+ ch^2\left(\|\mathbf{W}_h\|_{L^2_{L^2(\Gamma_h^0)}}\|\boldsymbol{\phi}_h\|_{L^2_{L^2(\Gamma_h^0)}}\right).\end{aligned} \tag{3.49}$$

The last two terms tend to zero as $h \to 0$. For the other terms we have just proved their weak convergence in (3.45-3.47). Hence the limit satisfies (3.26):

$$\int_0^T s(\hat{\mathbf{u}},\boldsymbol{\chi}_h) - m(\hat{\mathbf{w}},\boldsymbol{\chi}_h)dt = 0 \qquad \forall \boldsymbol{\chi}_h \in S_h(\Gamma^0) \tag{3.50}$$

Next we need strong convergence of $\nabla_{\Gamma^0}\mathbf{u}_h$, from (3.50) conclude the following relation

$$\int_0^T s(\hat{\mathbf{u}},\boldsymbol{\varphi}_h) - s_h(\mathbf{U}_h,\boldsymbol{\phi}_h)dt = \int_0^T m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h) - m_h(\mathbf{W}_h,\boldsymbol{\phi}_h)dt.$$

This can be expanded to include $\mathbf{u}_h$ so that

$$\begin{aligned}&\int_0^T s(\hat{\mathbf{u}}-\mathbf{u}_h,\boldsymbol{\varphi}_h)dt + \int_0^T \left(s(\mathbf{u}_h,\boldsymbol{\varphi}_h) - s_h(\mathbf{U}_h,\boldsymbol{\phi}_h)\right)dt \\ =&\int_0^T m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h) - m_h(\mathbf{W}_h,\boldsymbol{\phi}_h)dt.\end{aligned}$$

We can now make use of the convergence of Lemma 3.2.3

$$\begin{aligned}\int_0^T s(\hat{\mathbf{u}}-\mathbf{u}_h,\boldsymbol{\varphi}_h)dt &= \int_0^T m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h)-m_h(\mathbf{W}_h,\boldsymbol{\phi}_h)+\left(s_h(\mathbf{U}_h,\boldsymbol{\phi}_h)-s(\mathbf{u}_h,\boldsymbol{\varphi}_h)\right)dt\\ &\leq \int_0^T m(\hat{\mathbf{w}},\boldsymbol{\varphi}_h)-m(\mathbf{w}_h,\boldsymbol{\varphi}_h)+m(\mathbf{w}_h,\boldsymbol{\varphi}_h)-m_h(\mathbf{W}_h,\boldsymbol{\phi}_h)dt\\ &\quad + ch^2\int_0^T \|\nabla_{\Gamma^0}\mathbf{U}_h\|_{L^2(\Gamma_h^0)}\|\nabla_{\Gamma^0}\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}\,dt\\ &\leq \int_0^T m(\hat{\mathbf{w}}-\mathbf{w}_h,\boldsymbol{\varphi}_h)dt + ch^2\|\mathbf{W}_h\|_{L^2(\Gamma_h^0)}\|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}\\ &\quad + ch^2\int_0^T \|\nabla_{\Gamma^0}\mathbf{U}_h\|_{L^2(\Gamma_h^0)}\|\nabla_{\Gamma^0}\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}\,dt \qquad (3.51)\end{aligned}$$

We prove the strong convergence of the gradient $\nabla_{\Gamma^0}\hat{\mathbf{u}}$ by recalling the Ritz projection (Definition 3.2.3) and noting $\hat{\mathbf{u}}\in L^2_{H^1(\Gamma^0)}$, so can use (3.13). This allows us to state that $\pi_h\hat{\mathbf{u}}\to\hat{\mathbf{u}}$ in $L^2_{H^1(\Gamma^0)}$:

$$\begin{aligned}\|\nabla_{\Gamma^0}(\hat{\mathbf{u}}-\mathbf{u}_h)\|^2_{L^2_{L^2(\Gamma^0)}} &= \int_0^T s(\hat{\mathbf{u}}-\mathbf{u}_h,\hat{\mathbf{u}}-\mathbf{u}_h)dt\\ &= \int_0^T s(\hat{\mathbf{u}}-\pi_h\hat{\mathbf{u}}+\pi_h\hat{\mathbf{u}}-\mathbf{u}_h,\hat{\mathbf{u}}-\mathbf{u}_h)dt\\ &= \int_0^T s(\hat{\mathbf{u}}-\pi_h\hat{\mathbf{u}},\hat{\mathbf{u}}-\mathbf{u}_h)+s(\pi_h\hat{\mathbf{u}}-\mathbf{u}_h,\hat{\mathbf{u}}-\mathbf{u}_h)dt. \qquad (3.52)\end{aligned}$$

We can use identity (3.51) and grouping the $O(h^2)$ terms under Ch^2 to bound the second term on the right hand side of (3.52) as $\mathbf{u}_h - \pi_h \mathbf{u} \in S_h^3(\Gamma^0)$, whence as $h \to 0$.

$$
\begin{aligned}
&\|\nabla_{\Gamma^0}\hat{\mathbf{u}} - \mathbf{u}_h\|_{L^2_{L^2}} \\
&\leq \int_0^T s(\hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}}, \hat{\mathbf{u}} - \mathbf{u}_h) + m(\hat{\mathbf{w}} - \mathbf{w}_h, \mathbf{u}_h - \pi_h\hat{\mathbf{u}})dt + Ch^2 \\
&\leq \int_0^T s(\hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}}, \hat{\mathbf{u}} - \mathbf{u}_h) + m(\hat{\mathbf{w}} - \mathbf{w}_h, \mathbf{u}_h - \hat{\mathbf{u}} + \hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}})dt + Ch^2 \\
&\leq \int_0^T s(\hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}}, \hat{\mathbf{u}} - \mathbf{u}_h) + m(\hat{\mathbf{w}} - \mathbf{w}_h, \mathbf{u}_h - \hat{\mathbf{u}}) + m(\hat{\mathbf{w}} - \mathbf{w}_h, \hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}})dt + Ch^2 \\
&\leq \int_0^T \underbrace{\|\nabla_{\Gamma^0}(\hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}})\|_{L^2(\Gamma^0)}}_{=0 \text{ by (3.13)}} \underbrace{\left(\|\nabla_{\Gamma^0}\hat{\mathbf{u}}\|_{L^2(\Gamma^0)} + \|\nabla_{\Gamma^0}\mathbf{u}_h\|_{L^2(\Gamma^0)}\right)}_{\leq C \text{ by (3.42)}} dt \\
&\quad + \int_0^T \underbrace{\left(\|\hat{\mathbf{w}}\|_{L^2(\Gamma^0)} + \|\mathbf{w}_h\|_{L^2(\Gamma^0)}\right)}_{\leq C \text{ by 3.42}} \left(\underbrace{\|\mathbf{u}_h - \hat{\mathbf{u}}\|_{L^2(\Gamma^0)}}_{\to 0 \text{ (3.48)}} + \underbrace{\|\hat{\mathbf{u}} - \pi_h\hat{\mathbf{u}}\|_{L^2(\Gamma^0)}}_{\to 0 \text{ (3.13)}} \right) dt + Ch^2.
\end{aligned}
$$

Note $\nabla_{\Gamma^0}\hat{\mathbf{u}}$ is the weak limit of $\nabla_{\Gamma^0}\mathbf{u}_h$, which is uniformly bounded and the norm is weakly lower semi continuous so $\nabla_{\Gamma^0}\hat{\mathbf{u}}$ itself is bounded. Similarly $\hat{\mathbf{w}}$ is also the weak limit of $\nabla_{\Gamma^0}\mathbf{w}_h$ which is bounded in $L^2_{H^1(\Gamma^0)}$ so follows the same argument. The right hand side is now written in terms of bounds independant of h and known convergence results that tend to 0 as $h \to 0$. Hence we conclude that $\nabla_{\Gamma^0}\mathbf{u}_h \to \nabla_{\Gamma^0}\hat{\mathbf{u}}$ in $L^2_{L^2(\Gamma^0)}$.

Thanks to the growth properties of ψ_A (3.23), we have

$$|\psi_A(\nabla_{\Gamma^0}\mathbf{u}_h, \nabla_{\Gamma^0}\mathbf{u}_{0,h})| \leq C_\psi \, |\nabla_{\Gamma^0}\mathbf{u}_{0,h}| \, |\nabla_{\Gamma^0}\mathbf{u}_h| + \gamma_\psi,$$

independent of h. The strong convergence of the gradient implies that $\nabla_{\Gamma^0}\mathbf{u}_h \to \nabla_{\Gamma^0}\hat{\mathbf{u}}$ almost everywhere and for $\nabla_{\Gamma^0}\mathbf{u}_{0,h}$. Thus,

$$\psi_A(\nabla_{\Gamma^0}\mathbf{u}_h, \nabla_{\Gamma^0}\mathbf{u}_{0,h}) \to \psi_A(\nabla_{\Gamma^0}\hat{\mathbf{u}}, \nabla_{\Gamma^0}\hat{\mathbf{u}}_0),$$

holds almosts anywhere. As this is bounded we can apply Lebesgue's general convergence theorem A.0.1, and therefore have that

$$\psi_A(\nabla_{\Gamma^0}\mathbf{u}_h, \nabla_{\Gamma^0}\mathbf{u}_{0,h}) \to \psi_A(\nabla_{\Gamma^0}\hat{\mathbf{u}}, \nabla_{\Gamma^0}\hat{\mathbf{u}}_0) \qquad \text{strongly in } L^2_{L^2(\Gamma^0)}. \tag{3.53}$$

We now give a similar argument for $\mathbf{k}$. Using the Lipschitz condition from (3.20)

we have

$$\begin{aligned}
&\left|\mathbf{k}(\hat{\mathbf{u}},\mathbf{x},\boldsymbol{\nu}_{\Gamma^0})-\mathbf{k}_h(\mathbf{u}_h,\mathbf{x},\boldsymbol{\nu}^l_{\Gamma^0_h})\right|\\
\leq&\left|\mathbf{k}(\hat{\mathbf{u}},\mathbf{x},\boldsymbol{\nu}_{\Gamma^0})-\mathbf{k}(\mathbf{u}_h,\mathbf{x},\boldsymbol{\nu}^l_{\Gamma^0_h})+\mathbf{k}(\mathbf{u}_h,\mathbf{x},\boldsymbol{\nu}^l_{\Gamma^0_h})-\mathbf{k}_h(\mathbf{u}_h,\mathbf{x},\boldsymbol{\nu}^l_{\Gamma^0_h})\right|\\
\leq& C_k\left(\underbrace{|\hat{\mathbf{u}}-\mathbf{u}_h|}_{\to 0\ (3.48)}+\underbrace{\left|\boldsymbol{\nu}_{\Gamma^0}-\boldsymbol{\nu}^l_{\Gamma^0_h}\right|}_{\leq ch\ (3.6)}\right)+\underbrace{\|\mathbf{k}-\mathbf{k}_h\|_{L^\infty}}_{\leq ch(\|\mathbf{u}\|+1)\ (3.22)}.
\end{aligned}$$

3.8.2 Convergence

We have introduced the projection operator in (3.1), hence we know for $\boldsymbol{\varphi}\in L^2_{H^1(\Gamma^0)}$ there exists its Ritz projection to the discrete level denoted $\boldsymbol{\phi}_h=\Pi_h\boldsymbol{\varphi}$ (from Definition 3.2.3) and also we have $\boldsymbol{\varphi}_h=\boldsymbol{\phi}^l_h=\pi_h\boldsymbol{\varphi}$. In order to have better control over ψ_A however we will restrict to test functions $\boldsymbol{\varphi}\in L^2_{H^2(\Gamma^0)}$ for now. By integrating (3.27) in time we can write .

$$\begin{aligned}
&\int_0^T m(\partial_t\hat{\mathbf{u}},\boldsymbol{\varphi})+s(\hat{\mathbf{w}},\boldsymbol{\varphi})dt+\int_0^T s(\psi_A;\hat{\mathbf{u}},\hat{\mathbf{u}}_0,\boldsymbol{\varphi})dt+\int_0^T m(\mathbf{k}(\hat{\mathbf{u}},\mathbf{x},\boldsymbol{\nu}_{\Gamma^0}),\boldsymbol{\varphi})dt\\
=&\int_0^T\left(m(\partial_t\hat{\mathbf{u}},\boldsymbol{\varphi})-m_h(\partial_t\mathbf{U}_h,\boldsymbol{\phi}_h)\right)dt+\int_0^T\left(s(\hat{\mathbf{w}},\boldsymbol{\varphi})-s_h(\mathbf{W}_h,\boldsymbol{\phi}_h)\right)dt\\
&+\int_0^T\left(s(\psi_A;\hat{\mathbf{u}},\hat{\mathbf{u}}_0,\boldsymbol{\varphi})-s_h(\psi_A;\mathbf{U}_h,\mathbf{U}^0_h,\boldsymbol{\phi}_h)\right)dt\\
&+\int_0^T\left(m(\mathbf{k}(\hat{\mathbf{u}},\mathbf{x},\boldsymbol{\nu}^0_{\Gamma^0_h}),\boldsymbol{\varphi})-m_h(\mathbf{k}_h(\mathbf{U}_h,\mathbf{X}_h,\boldsymbol{\nu}^0_{\Gamma^0_h}),\boldsymbol{\phi}_h)\right)dt\\
=:&I_1+I_2+I_3+I_4.
\end{aligned}$$

We consider each term in the following separately. We split I_1 as follows

$$\begin{aligned}
I_1=&\int_0^T m(\partial_t\hat{\mathbf{u}}-\partial_t\mathbf{u}_h,\boldsymbol{\varphi})+m(\partial_t\mathbf{u}_h,\boldsymbol{\varphi}-\boldsymbol{\varphi}_h)+\left(m(\partial_t\mathbf{u}_h,\boldsymbol{\varphi}_h)-m_h(\partial_t\mathbf{U}_h,\boldsymbol{\phi}_h)\right)dt\\
=:&\,I_{1,1}+I_{1,2}+I_{1,3}.
\end{aligned}$$

We know that $I_{1,1}\to 0$ as $h\to 0$ by the convergence result (3.48). The second term can be bounded by using the convergence of the Ritz projection (3.13) and the final

term is an application of the convergence results (Lemma 3.2.3) hence we obtain

$$I_{1,2} \leq \int_0^T \|\partial_t \hat{\mathbf{u}}\|_{L^2(\Gamma^0)} \|\boldsymbol{\varphi} - \boldsymbol{\varphi}_h\|_{L^2(\Gamma^0)} \, dt \leq ch^2 \int_0^T \|\partial_t \hat{\mathbf{u}}\|_{L^2(\Gamma^0)} \|\boldsymbol{\varphi}\|_{H^1(\Gamma^0)} \, dt,$$
$$I_{1,3} \leq ch^2 \int_0^T \|\partial_t \mathbf{U}_h\|_{L^2(\Gamma_h^0)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)} \, dt \leq ch^2 \left(\|\partial_t \hat{\mathbf{u}}\|_{L^2_{L^2}} \|\boldsymbol{\varphi}\|_{L^2_{L^2}} \right).$$

A similar argument can be used for I_2. We write

$$\begin{aligned} I_2 &= \int_0^T s(\hat{\mathbf{w}} - \mathbf{w}_h, \boldsymbol{\varphi}) + (s(\mathbf{w}_h, \boldsymbol{\varphi}) - s_h(\mathbf{W}_h, \boldsymbol{\phi}_h)) \, dt \\ &=: I_{2,1} + I_{2,2}. \end{aligned}$$

Convergence of $I_{2,1}$ follows by result (3.47). From the Ritz projection (Definition 3.2.3) we have $I_{2,2} = 0$. Now we consider the nonlinear term ψ_A:

$$\begin{aligned} I_3 = &\int_0^T \left[\int_{\Gamma^0} (\psi_A(\nabla_{\Gamma^0} \hat{\mathbf{u}}, \nabla_{\Gamma^0} \hat{\mathbf{u}}_0) - \psi_A(\nabla_{\Gamma^0} \mathbf{u}_h, \nabla_{\Gamma^0} \hat{\mathbf{u}}_{0,h})) \nabla_{\Gamma^0} \boldsymbol{\varphi} d\sigma \right] dt \\ &+ \int_0^T [s(\psi_A; \mathbf{u}_h, \mathbf{u}_{0,h}, \boldsymbol{\varphi} - \boldsymbol{\varphi}_h)] \, dt \\ &+ \int_0^T \left[s(\psi_A; \mathbf{u}_h, \mathbf{u}_{0,h}, \boldsymbol{\varphi}_h) - s_h(\psi_A; \mathbf{U}_h, \mathbf{U}_h^0, \boldsymbol{\phi}_h) \right] dt \\ =: &I_{3,1} + I_{3,2} + I_{3,3}. \end{aligned} \tag{3.54}$$

Convergence of the first part $I_{3,1}$ follows by (3.53). Since $\boldsymbol{\phi} \in L^2_{H^2(\Gamma_h^0)}$ we can use (3.14) and the growth assumption (3.23) to obtain that

$$\begin{aligned} I_{3,2} \leq &ch \int_0^T \left(\left\| \nabla_{\Gamma_h^0} \mathbf{U} \right\|_{L^2(\Gamma_h^0)} + 1 \right) \|\boldsymbol{\phi}_h\|_{H^2(\Gamma_h^0)} \, dt \\ \leq &ch \int_0^T \left(\|\nabla_{\Gamma^0} \hat{\mathbf{u}}\|_{L^2(\Gamma^0)} + 1 \right) \|\boldsymbol{\varphi}\|_{H^2(\Gamma^0)} \, dt. \end{aligned}$$

The final term is more technical. Writing $\nabla_{\Gamma_h^0} \boldsymbol{\phi}_h$ explicitly in terms of its lift $\nabla_{\Gamma^0} \boldsymbol{\varphi}_h$, using notation from Lemma 3.1.2, that $\nabla_{\Gamma_h^0} \boldsymbol{\phi}_h = \mathbf{Q}_h \nabla_{\Gamma^0} \boldsymbol{\varphi}_h$, here $\mathbf{Q}_h = \mathbf{P}_h(\mathbf{I} - d\mathcal{H})\mathbf{P}$. Similarly for $\mathbf{U}_h$ and $\mathbf{U}_h^0$ can also be explicitly written in terms

defined over Γ^0 instead of Γ_h^0. Combining these we obtain

$$\begin{aligned}
I_{3,3} =& \int_0^T \left[\int_{\Gamma^0} \psi_A \left(\nabla_{\Gamma^0} \mathbf{u}_h, \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) : \nabla_{\Gamma^0} \boldsymbol{\varphi}_h d\sigma \right] dt \\
& - \int_0^T \left[\int_{\Gamma^0} \psi_A \left(\mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h, \mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) : \mathbf{Q}_h \nabla_{\Gamma^0} \boldsymbol{\varphi}_h \frac{1}{\delta_h} d\sigma \right] dt \\
=& \int_0^T \left[\int_{\Gamma^0} \left(\psi_A \left(\nabla_{\Gamma^0} \mathbf{u}_h, \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) - \psi_A \left(\mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h, \mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) \right) : \nabla_{\Gamma^0} \boldsymbol{\varphi}_h d\sigma \right] dt \\
& + \int_0^T \left[\int_{\Gamma^0} \psi_A \left(\mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h, \mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) : \left(\mathbf{P} - \frac{1}{\delta_h} \mathbf{Q}_h \right) \nabla_{\Gamma^0} \boldsymbol{\varphi}_h d\sigma \right] dt \\
=& I_{3,3,1} + I_{3,3,2}.
\end{aligned}$$

For $I_{3,3,1}$, it follows by the Lipschitz condition in Assumption 1 which allows us to then apply (3.7) and (3.42)

$$\begin{aligned}
I_{3,3,1} \leq & C_\psi \left(\left\| \left(\mathbf{P} - \mathbf{Q}_h \right) \nabla_{\Gamma^0} \ \mathbf{u}_h \right\|_{L^2_{L^2(\Gamma^0)}}^2 + \left\| \left(\mathbf{P} - \mathbf{Q}_h \right) \nabla_{\Gamma^0} \ \mathbf{u}_h^0 \right\|_{L^2_{L^2(\Gamma^0)}}^2 \right) \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi}_h \right\|_{L^2_{L^2(\Gamma^0)}}^2 \\
\leq & ch \left(\left\| \nabla_{\Gamma^0} \ \mathbf{u}_h \right\|_{L^2_{L^2(\Gamma^0)}}^2 + \left\| \nabla_{\Gamma^0} \ \mathbf{u}_h^0 \right\|_{L^2_{L^2(\Gamma^0)}}^2 \right) \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi}_h \right\|_{L^2_{L^2(\Gamma^0)}} \\
\leq & ch \left(\left\| \nabla_{\Gamma^0} \hat{\mathbf{u}} \right\|_{L^2_{L^2(\Gamma^0)}}^2 + 1 \right) \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi} \right\|_{L^2_{L^2(\Gamma^0)}}.
\end{aligned}$$

Now for the second part $I_{3,3,2}$ we use (3.7) and (3.23) to get

$$\begin{aligned}
I_{3,3,2} \leq & \left\| \mathbf{Q}_h \frac{1}{\delta_h} - \mathbf{P} \right\|_{L^\infty(\Gamma^0)} \left\| \psi_A \left(\mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h, \mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) \right\|_{L^2_{L^2(\Gamma^0)}} \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi}_h \right\|_{L^2_{L^2(\Gamma^0)}} \\
\leq & ch \left\| \psi_A \left(\mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h, \mathbf{Q}_h \nabla_{\Gamma^0} \mathbf{u}_h^0 \right) \right\|_{L^2_{L^2(\Gamma^0)}} \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi}_h \right\|_{L^2_{L^2(\Gamma^0)}} \\
\leq & ch \left(\left\| \nabla_{\Gamma^0} \mathbf{u}_h \right\|_{L^2_{L^2(\Gamma^0)}}^2 + 1 \right) \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi}_h \right\|_{L^2_{L^2(\Gamma^0)}} \\
\leq & ch \left(\left\| \nabla_{\Gamma^0} \hat{\mathbf{u}} \right\|_{L^2_{L^2(\Gamma^0)}}^2 + 1 \right) \left\| \nabla_{\Gamma^0} \boldsymbol{\varphi} \right\|_{L^2_{L^2(\Gamma^0)}}.
\end{aligned}$$

In the penultimate step we assume $\mathbf{Q}_h$ is bounded which holds for sufficiently small h. Finally we have the lowest order term.

$$\begin{aligned}
I_4 =& \int_0^T \left(m(\mathbf{k}(\hat{\mathbf{u}}, \mathbf{x}_h, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}) - m(\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0_h}), \boldsymbol{\varphi}) \right) dt \\
&+ \int_0^T m(\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0_h}), \boldsymbol{\varphi} - \boldsymbol{\varphi}_h) dt \\
&+ \int_0^T m(\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0_h}), \boldsymbol{\varphi}_h) - m(\mathbf{k}_h(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0_h}), \boldsymbol{\varphi}_h) dt \\
&+ \int_0^T m(\mathbf{k}_h(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0_h}), \boldsymbol{\varphi}_h) - m_h(\mathbf{k}_h(\mathbf{U}_h, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma^0_h}), \boldsymbol{\phi}_h) dt \\
=& I_{4,1} + I_{4,2} + I_{4,3} + I_{4,4}.
\end{aligned}$$

As $\mathbf{k}$ is continuous in its arguments (Assumption 2) we have convergence of $I_{4,1} \to 0$ thanks to (3.48) and (3.6). For $I_{4,3}$ we know that convergence is at worst $o(h)$ by assumption (3.22) over the larger domain D_δ so this must hold over Γ^0.

$$\begin{aligned}
I_{4,1} =& \int_0^T \underbrace{m(\mathbf{k}(\hat{\mathbf{u}}, \mathbf{x}_h, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}) - m(\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi})}_{\to 0 \text{ as } h \to 0} dt, \\
&+ \int_0^T \underbrace{m(\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}) - m(\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0}), \boldsymbol{\varphi})}_{\to 0 \text{ as } h \to 0} dt, \\
I_{4.2} =& \int_0^T \|\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}_{\Gamma^0})\|_{L^2(\Gamma^0)} \|\boldsymbol{\varphi} - \boldsymbol{\varphi}_h\|_{L^2(\Gamma^0)} dt \\
\leq& ch \int_0^T \|\mathbf{k}(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}_{\Gamma^0})\|_{L^2(\Gamma^0)} \|\boldsymbol{\varphi}\|_{H^1(\Gamma^0)} dt \\
\leq& Ch \left(\|\mathbf{u}_h\|^2_{L^2_{L^2(\Gamma^0)}} + 1 \right) \|\boldsymbol{\varphi}\|^2_{L^2_{H^1(\Gamma^0)}}, \\
I_{4,3} \leq& ch \int_0^T \left(\|\mathbf{u}_h\|_{L^2(\Gamma^0)} + 1 \right) \|\boldsymbol{\varphi}\|_{L^2(\Gamma^0)} dt \leq Ch \left(\|\mathbf{u}_h\|^2_{L^2_{L^2(\Gamma^0)}} + 1 \right) \|\boldsymbol{\varphi}\|^2_{L^2_{L^2(\Gamma^0)}}, \\
I_{4,4} \leq& \int_0^T m\left(\left(1 - \frac{1}{\delta^l_h} \right) \mathbf{k}_h(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0}), \boldsymbol{\varphi} \right) dt \\
&+ \int_0^T m\left(\frac{1}{\delta^l_h} \mathbf{k}_h(\mathbf{u}_h, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0}), \boldsymbol{\varphi} \right) - m\left(\frac{1}{\delta^l_h} \mathbf{k}_h(\mathbf{u}_h, \mathbf{x}_h, \mathbf{Q}_h \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi} \right) dt \\
\leq& c \int_0^T h^2 \|\mathbf{U}_h\|_{L^2(\Gamma^0_h)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma^0_h)} + h \|\mathbf{U}_h\|_{L^2(\Gamma^0_h)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma^0_h)} dt \\
\leq& C \left(h^2 \|\hat{\mathbf{u}}\|^2_{L^2_{L^2(\Gamma^0)}} \|\boldsymbol{\varphi}\|^2_{L^2_{L^2(\Gamma^0)}} + h \|\hat{\mathbf{u}}\|^2_{L^2_{L^2(\Gamma^0)}} \|\boldsymbol{\varphi}\|^2_{L^2_{L^2(\Gamma^0)}} \right).
\end{aligned}$$

All together we thus conclude that in the limit as $h \to 0$ the weak limits $\hat{\mathbf{u}}$

and $\hat{\mathbf{w}}$ solve the follow equation

$$\int_0^T m(\partial_t \hat{\mathbf{u}}, \boldsymbol{\varphi}) + s(\hat{\mathbf{w}}, \boldsymbol{\varphi}) + s(\psi_A; \hat{\mathbf{u}}, \mathbf{u}_0, \boldsymbol{\varphi}) + m(\mathbf{k}(\hat{\mathbf{u}}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}) dt \quad \forall \boldsymbol{\varphi} \in L^2_{H^2(\Gamma^0)}.$$

We can now combine this with (3.50). By density of H^2 in H^1, we can conclude that for all $\boldsymbol{\varphi}, \boldsymbol{\chi} \in L^2_{H^1}$

$$\int_0^T m(\partial_t \hat{\mathbf{u}}, \boldsymbol{\varphi}) + s(\hat{\mathbf{w}}, \boldsymbol{\varphi}) + s(\psi_A; \hat{\mathbf{u}}, \mathbf{u}_0, \boldsymbol{\varphi}) + m(\mathbf{k}(\hat{\mathbf{u}}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}) = 0,$$
$$\int_0^T m(\hat{\mathbf{w}}, \boldsymbol{\chi}) - s(\hat{\mathbf{u}}, \boldsymbol{\chi}) = 0.$$

As this holds for almost all $t \in (0, T)$ we can drop the time integrals by Lemma 2.1.3 and have that $(\hat{\mathbf{u}}, \hat{\mathbf{w}})$ is a solution to equations (3.25)(3.26), it remains to show uniqueness and that it satisfies initial conditions. For the later we have for all $\boldsymbol{\varphi} \in C^1_{H^1}$ that decay at final time T, denoted by $\boldsymbol{\varphi}(T) = \mathbf{0}$ and that $\mathbf{U}_h^0 := \Pi_h \mathbf{u}_0$ then by (3.13) we have as $h \to 0$

$$m_h(\mathbf{U}_h^0, \boldsymbol{\phi}_h(0)) \to m(\mathbf{u}_0, \boldsymbol{\varphi}(0)).$$

Using $\boldsymbol{\phi}_h(T) = \mathbf{0}$ this is equal to

$$\begin{aligned} m_h(\mathbf{U}_h^0, \boldsymbol{\phi}_h(0)) =& m_h(\mathbf{U}_h^0, \boldsymbol{\phi}_h(0)) - m_h(\mathbf{U}_h(T), \boldsymbol{\phi}_h(T)) \\ =& -\int_0^T \partial_t m_h(\mathbf{U}_h, \boldsymbol{\phi}_h) dt \\ =& -\int_0^T m_h(\partial_t \mathbf{U}_h, \boldsymbol{\phi}_h) + m_h(\mathbf{U}_h, \partial_t \boldsymbol{\phi}_h) dt. \end{aligned}$$

Which in the limit as $h \to 0$ we know from the proof that discrete limit $\hat{\mathbf{u}}$ satisfies the continuum equations (3.25) and (3.26), so becomes

$$\begin{aligned} -\int_0^T m_h(\partial_t \mathbf{U}_h, \boldsymbol{\phi}_h) + m_h(\mathbf{U}_h, \partial_t \boldsymbol{\phi}_h) dt \to& -\int_0^T m_h(\partial_t \hat{\mathbf{u}}, \boldsymbol{\varphi}) + m(\hat{\mathbf{u}}, \partial_t \boldsymbol{\varphi}) dt \\ =& m(\hat{\mathbf{u}}(0), \boldsymbol{\varphi}(0)) - m(\hat{\mathbf{u}}(T), \boldsymbol{\varphi}(T)) \\ =& m(\hat{\mathbf{u}}(0), \boldsymbol{\varphi}(0)). \end{aligned}$$

We conclude $\hat{\mathbf{u}}(\cdot, 0) = \mathbf{u}_0$ almost everywhere in Γ^0.

3.8.3 Uniqueness

Take the difference of two solutions $\boldsymbol{\xi}^{\mathbf{u}} := \mathbf{u}^1 - \mathbf{u}^2$ and $\boldsymbol{\xi}^{\mathbf{w}} := \mathbf{w}^1 - \mathbf{w}^2$. Comparing both solutions gives for all $\boldsymbol{\phi}, \boldsymbol{\chi} \in [H^1(\Gamma^0)]^3$

$$m(\partial_t \boldsymbol{\xi}^{\mathbf{u}}, \boldsymbol{\phi}) + s(\boldsymbol{\xi}^{\mathbf{w}}, \boldsymbol{\phi}) + s(\psi_A; \mathbf{u}^1, \mathbf{u}^0, \boldsymbol{\phi}) - s(\psi_A; \mathbf{u}^2, \mathbf{u}^0, \boldsymbol{\phi}) + m(\mathbf{k}(\mathbf{u}^1, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}) - \mathbf{k}(\mathbf{u}^2, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\phi}) = 0, \tag{3.55}$$

$$m(\boldsymbol{\xi}^{\mathbf{w}}, \boldsymbol{\chi}) = s(\boldsymbol{\xi}^{\mathbf{u}}, \boldsymbol{\chi}). \tag{3.56}$$

Now we taking $\boldsymbol{\phi} = \boldsymbol{\xi}^{\mathbf{u}}$ in the first equation (3.55) and $\boldsymbol{\chi} = \boldsymbol{\xi}^{\mathbf{w}}$ in the second equation (3.56). We use that $\mathbf{k}$ and ψ_A are Lipschitz due to Assumption 1 and 2. Hence we have

$$\frac{d}{dt} \|\boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 + \|\boldsymbol{\xi}^{\mathbf{w}}\|_{L^2}^2 \leq 2C_k \|\boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 + C_\psi \|\nabla_{\Gamma^0} \boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 . \tag{3.57}$$

For the second equation a similar argument gives for $\epsilon > 0$

$$\|\nabla_{\Gamma^0} \boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 \leq \frac{1}{2\epsilon} \|\boldsymbol{\xi}^{\mathbf{w}}\|_{L^2}^2 + \frac{\epsilon}{2} \|\boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 .$$

This allows us to bound the gradient on right hand side of (3.57) and then move the $\boldsymbol{\xi}^{\mathbf{w}}$ to the left by taking the weight to be the Lipschitz constant $\epsilon = C_\psi$.

$$\frac{d}{dt} \|\boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 + \frac{1}{2} \|\boldsymbol{\xi}^{\mathbf{w}}\|_{L^2}^2 \leq \left(2C_k + \frac{C_\psi}{2}\right) \|\boldsymbol{\xi}^{\mathbf{u}}\|_{L^2}^2 . \tag{3.58}$$

Integrating in time gives for some constant λ

$$\|\boldsymbol{\xi}^{\mathbf{u}}\|_{L^2} \leq \left(2C_k + \frac{C_\psi}{2}\right) e^{\lambda t} \left\|\mathbf{u}^1(0) - \mathbf{u}^2(0)\right\|_{L^2} ,$$

which show uniqueness as solution must agree on initial condition and also the continuous dependence on them. As the right hand side of (3.58) does not depend on $\mathbf{w}$ we can conclude similarly that $\boldsymbol{\xi}^{\mathbf{w}} = 0$. □

3.9 Error convergence

For this next section we assume $\mathbf{u}, \partial_t \mathbf{u} \in L^2_{H^2(\Gamma^0)}$, as we require extra regularity to quantify the error rates by using Theorem 3.13. If this holds then we get $\mathbf{w} \in L^2_{H^2(\Gamma^0)}$

for free because (3.26) can be written as

$$s(\mathbf{w},\boldsymbol{\varphi}) + m(\mathbf{w},\boldsymbol{\varphi}) = m(\mathbf{f},\boldsymbol{\varphi}) \qquad\qquad \forall \boldsymbol{\varphi} \in [H^1(\Gamma^0)]^3,$$

by writing the known $\mathbf{u}$ terms as $\mathbf{f}$ which by assumption is a function in L^2. This gives a standard variation problem that we can use known regularity results for.

To prove convergence of the error, we follow the standard approach and divide/split it into interpolation θ and discrete ρ errors. Using the Ritz projection Π_h

$$\begin{aligned}\rho^{\mathbf{u}} + \theta^{\mathbf{u}} &:= \mathbf{u}^{-l}(\cdot, \tau m) - \mathbf{U}_h = (\mathbf{u}^{-l} - \Pi_h \mathbf{u}) + (\Pi_h \mathbf{u} - \mathbf{U}_h),\\ \rho^{\mathbf{w}} + \theta^{\mathbf{w}} &:= \mathbf{w}^{-l} - \mathbf{W}_h = (\mathbf{w}^{-l} - \Pi_h \mathbf{w}) + (\Pi_h \mathbf{w} - \mathbf{W}_h).\end{aligned}$$

We can use ideas from the convergence proof in the previous section to find bounds for ρ. With the regularity assumption we can now use Theorem 3.13.

The second part is now to bound θ. Throughout this section we write $\boldsymbol{\varphi}_h = \boldsymbol{\phi}_h^l$. Writing Problem 3.3.1 minus Problem 3.3.2 and inserting Π_h we can rearrange the equation to separate θ terms on the left and ρ on the right and hence obtain

$$\begin{aligned}&m_h(\partial_t \theta^{\mathbf{u}}, \boldsymbol{\phi}_h) + s_h(\theta^{\mathbf{w}}, \boldsymbol{\phi}_h) + s_h(\psi_A; \Pi_h \mathbf{u}, \mathbf{U}_h^0, \boldsymbol{\phi}_h) - s_h(\psi_A; \mathbf{U}_h, \mathbf{U}_h^0, \boldsymbol{\phi}_h)\\ &\quad + m_h(\mathbf{k}_h(\Pi_h \mathbf{u}, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h) - m_h(\mathbf{k}_h(\mathbf{U}_h, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h)\\ =&(m_h(\partial_t \Pi_h \mathbf{u}, \boldsymbol{\phi}_h) - m(\partial_t \mathbf{u}, \boldsymbol{\varphi}_h)) + (s_h(\psi_A; \Pi_h \mathbf{u}, \mathbf{U}_h^0, \boldsymbol{\phi}_h) - s(\psi_A; \mathbf{u}, \mathbf{u}_0, \boldsymbol{\varphi}_h))\\ &\quad + m_h(\mathbf{k}_h(\Pi_h \mathbf{u}, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h) - m(\mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}_h)\\ =:&E_t(\boldsymbol{\phi}_h) + E_\psi(\boldsymbol{\phi}_h) + E_k(\boldsymbol{\phi}_h).\end{aligned} \tag{3.59}$$

For the second equation we then have

$$s_h(\theta^{\mathbf{u}}, \boldsymbol{\phi}_h) - m_h(\theta^{\mathbf{w}}, \boldsymbol{\phi}_h) = (m_h(\Pi_h \mathbf{w}, \boldsymbol{\phi}_h) - m(\mathbf{w}, \boldsymbol{\varphi}_h)) =: E_w(\boldsymbol{\phi}_h). \tag{3.60}$$

We pick up no gradient terms because of the Definition of Ritz projection in Theorem 3.2.3.

Lemma 3.9.1. *For our problem we have that for sufficiently small h and all $\boldsymbol{\phi}_h \in S_h^3$*

$$|E_t(\boldsymbol{\phi}_h)| \leq ch^2 \|\partial_t \mathbf{u}\|_{H^2(\Gamma^0)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}, \tag{3.61}$$

$$|E_\psi(\boldsymbol{\phi}_h)| \leq ch \|\mathbf{u}\|_{H^2(\Gamma^0)} \left\|\nabla_{\Gamma_h^0} \boldsymbol{\phi}_h\right\|_{L^2(\Gamma_h^0)}, \tag{3.62}$$

$$|E_k(\boldsymbol{\phi}_h)| \leq c(h + h^2 \|\mathbf{u}\|_{H^2(\Gamma^0)}) \|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}, \tag{3.63}$$

$$|E_w(\boldsymbol{\phi}_h)| \leq ch^2 \|\mathbf{w}\|_{H^2(\Gamma^0)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}, \tag{3.64}$$

where $c > 0$ is independent of h.

Proof. This proof follows a very similar structure as the convergence part of the proof for Theorem 3.8.1. Here we give only an overview of the steps. The first estimate is an application of (3.15)

$$\begin{aligned}|E_t(\boldsymbol{\phi}_h)| \leq& (m_h(\partial_t \Pi_h \mathbf{u}, \boldsymbol{\phi}_h) - m(\partial_t \mathbf{u}, \boldsymbol{\varphi}_h)) \\ \leq& ch^2 \|\partial_t \mathbf{u}\|_{H^2(\Gamma^0)} \|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}.\end{aligned}$$

For the non linear gradient term we can use that it is Lipschitz. This allows us to bound the term by something that can be treated by (3.14). The argument is similar to the one used for I_3 in (3.54).

First we separate out the geometric part, then the remainder can be bounded by the Lipschitz assumption (1).

$$\begin{aligned}|E_\psi(\boldsymbol{\phi}_h)| \leq& |s_h(\psi_A; \Pi_h \mathbf{u}, \mathbf{U}_h^0, \boldsymbol{\phi}_h) - s(\psi_A; \pi_h \mathbf{u}, \pi_h \mathbf{u}_0, \boldsymbol{\varphi}_h)| \\ &+ |s(\psi_A; \pi_h \mathbf{u}, \pi_h \mathbf{u}_0, \boldsymbol{\varphi}_h) - s(\psi_A; \mathbf{u}, \mathbf{u}_0, \boldsymbol{\varphi}_h)| \\ \leq& |s_h(\psi_A; \Pi_h \mathbf{u}, \mathbf{U}_h^0, \boldsymbol{\phi}_h) - s(\psi_A; \pi_h \mathbf{u}, \pi_h \mathbf{u}_0, \boldsymbol{\varphi}_h)| \\ &+ C_\psi \left\|\nabla_{\Gamma_h^0} \mathbf{u}_0\right\|_{L^2} \|\nabla_{\Gamma^0}(\mathbf{u} - \pi_h \mathbf{u})\|_{L^2(\Gamma^0)} \|\nabla_{\Gamma^0} \boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)} \\ &+ C_\psi \left\|\nabla_{\Gamma_h^0} \mathbf{u}\right\|_{L^2} \|\nabla_{\Gamma^0}(\mathbf{u}_0 - \pi_h \mathbf{u}_0)\|_{L^2(\Gamma^0)} \|\nabla_{\Gamma^0} \boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}.\end{aligned}$$

The first term can be handled using similar argument to $I_{3,3}$ in (3.54), instead of $\mathbf{U}_h$ and its lift the term contains $\Pi_h \mathbf{u}$ and its lift. This means we use the small h requirement from $I_{3,3,2}$. Note that norms are equivalent between Γ_h^0 and Γ^0 by Lemma 3.1.1 so we chose to bound over Γ_h^0 to match the left hand side of (3.59).

$$\begin{aligned}|E_\psi(\boldsymbol{\phi}_h)| \leq& ch(\|\mathbf{u}\|_{H^2} + 1) \|\boldsymbol{\phi}_h\|_{H^1} \\ &+ C_\psi h \left(\left\|\nabla_{\Gamma_h^0} \mathbf{u}\right\|_{L^2} \|\mathbf{u}_0\|_{H^2(\Gamma^0)} + \left\|\nabla_{\Gamma_h^0} \mathbf{u}_0\right\|_{L^2} \|\mathbf{u}\|_{H^2(\Gamma^0)}\right) \|\nabla_{\Gamma^0} \boldsymbol{\phi}_h\|_{L^2} \\ \leq& ch \|\mathbf{u}\|_{H^2(\Gamma^0)} \|\boldsymbol{\phi}_h\|_{H^1(\Gamma_h^0)}.\end{aligned}$$

For the third identity we use similar argument to above.

$$\begin{aligned}
|E_k(\boldsymbol{\phi}_h)| \leq & |m_h(\mathbf{k}_h(\Pi_h\mathbf{u}, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h) - m(\mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}_h)| \\
\leq & |m_h(\mathbf{k}_h(\Pi_h\mathbf{u}, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h) - m(\mathbf{k}_h(\pi_h\mathbf{u}, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma^0}), \boldsymbol{\varphi}_h)| \\
& + |m(\mathbf{k}_h(\pi_h\mathbf{u}, \mathbf{x}_h, \boldsymbol{\nu}^l_{\Gamma_h^0}), \boldsymbol{\varphi}_h) - m(\mathbf{k}(\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\varphi}_h)| \\
\leq & |m_h(\mathbf{k}_h(\Pi_h\mathbf{u}, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \boldsymbol{\phi}_h) - m(\mathbf{k}_h(\pi_h\mathbf{u}, \mathbf{x}, \boldsymbol{\nu}_{\Gamma^0}), \boldsymbol{\varphi}_h)| \\
& + C\left(h^2(\|\mathbf{u}\|_{L^2(\Gamma^0)} + 1) + h\|\boldsymbol{\nu}_{\Gamma^0}\|_{L^2(\Gamma^0)}\right)\|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)} \\
\leq & C\left(h^2\|\pi_h\mathbf{u}\|_{L^2(\Gamma^0)} + ch(\|\boldsymbol{\nu}_{\Gamma^0}\|_{L^2(\Gamma^0)} + 1)\right)\|\boldsymbol{\varphi}_h\|_{L^2(\Gamma^0)} + C(h^2 + h)\|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)} \\
\leq & C(h^2 + h)\|\boldsymbol{\phi}_h\|_{L^2(\Gamma_h^0)}.
\end{aligned}$$

□

Theorem 3.9.1 (Error analysis for FE scheme)**.** *Let* $\mathbf{u}, \mathbf{w}$ *solve Problem 2.5.1 and* $\mathbf{U}_h, \mathbf{W}_h$ *solve the semi discrete Problem 3.3.2. Assuming* $\mathbf{u}, \partial_t\mathbf{u} \in L^2_{H^2(\Gamma^0)}$ *and for sufficiently small* h *then we have that*

$$\sup_{t\in(0,T)}\left\|\mathbf{u}^{-l} - \mathbf{U}_h\right\|^2_{L^2(\Gamma_h^0)} + \int_0^T \left\|\mathbf{w}^{-l} - \mathbf{W}_h\right\|^2_{L^2(\Gamma_h^0)} + \left\|\nabla_{\Gamma_h^0}(\mathbf{u}^{-l} - \mathbf{U}_h)\right\|^2_{L^2(\Gamma_h^0)} dt \leq Ch^2.$$

Proof. We can now take $\boldsymbol{\phi}_h = \theta^{\mathbf{u}} \in S_h^3$ in the first equation (3.59) and $\boldsymbol{\phi}_h = \theta^{\mathbf{w}} \in S_h^3$ in the second (3.60). Taking the difference of both equations gives

$$\begin{aligned}
m_h(\partial_t\theta^{\mathbf{u}}, \theta^{\mathbf{u}}) + m_h(\theta^{\mathbf{w}}, \theta^{\mathbf{w}}) = & E_t(\theta^{\mathbf{u}}) + E_\psi(\theta^{\mathbf{u}}) + E_k(\theta^{\mathbf{u}}) - E_w(\theta^{\mathbf{w}}) \\
& + s_h(\psi_A; \Pi_h\mathbf{u}, \mathbf{U}_h^0, \theta^{\mathbf{u}}) - s_h(\psi_A; \mathbf{U}_h, \mathbf{U}_h^0, \theta^{\mathbf{u}}) \\
& + m_h(\mathbf{k}_h(\Pi_h\mathbf{u}, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}) - \mathbf{k}_h(\mathbf{U}_h, \mathbf{X}_h, \boldsymbol{\nu}_{\Gamma_h^0}), \theta^{\mathbf{u}}).
\end{aligned}$$

The function ψ_A is Lipschitz so we can bound it by the gradient. For lower order terms we have weaker convergence as E_k involves the unit normal. Altogether we have

$$\begin{aligned}
\frac{1}{2}\frac{d}{dt}m_h(\theta^{\mathbf{u}}, \theta^{\mathbf{u}}) + \|\theta^{\mathbf{w}}\|^2_{L^2} \leq & C_\psi\left\|\nabla_{\Gamma_h^0}\theta^{\mathbf{u}}\right\|^2_{L^2} + C_k\|\theta^{\mathbf{u}}\|^2_{L^2} \\
& + |E_t(\theta^{\mathbf{u}})| + |E_\psi(\theta^{\mathbf{u}})| + |E_k(\theta^{\mathbf{u}})| + |E_w(\theta^{\mathbf{w}})|.
\end{aligned}$$

Inserting $\boldsymbol{\phi}_h = \theta^{\mathbf{u}}$ in (3.60) gives

$$\left\|\nabla_{\Gamma_h^0}\theta^{\mathbf{u}}\right\|^2_{L^2} \leq \frac{1}{2}\|\theta^{\mathbf{w}}\|^2_{L^2} + \frac{1}{2}\|\theta^{\mathbf{u}}\|^2_{L^2} + E_w(\theta^{\mathbf{u}}), \tag{3.65}$$

which lets us suppress the dependence on $\nabla_{\Gamma^0}\theta^{\mathbf{u}}$.

$$\begin{aligned}\frac{1}{2}\frac{d}{dt}m_h(\theta^{\mathbf{u}},\theta^{\mathbf{u}})+\frac{1}{2}\left\|\theta^{\mathbf{w}}\right\|_{L^2}^2 \leq & C\left\|\theta^{\mathbf{u}}\right\|_{L^2}^2+\left|E_t(\theta^{\mathbf{u}})\right|+\left|E_\psi(\theta^{\mathbf{u}})\right| \\ & +\left|E_k(\theta^{\mathbf{u}})\right|+\left|E_w(\theta^{\mathbf{w}})\right|+\left|E_w(\theta^{\mathbf{u}})\right|.\end{aligned}$$

Now we also use the bounds from Lemma 3.9.1

$$\begin{aligned}\frac{d}{dt}m_h(\theta^{\mathbf{u}},\theta^{\mathbf{u}})+\left\|\theta^{\mathbf{w}}\right\|_{L^2}^2 \leq & C\left\|\theta^{\mathbf{u}}\right\|_{L^2}^2+\underbrace{c_t h^2\left\|\partial_t\mathbf{u}\right\|_{H^2}\left\|\theta^{\mathbf{u}}\right\|_{L^2}}_{\text{by (3.61)}} \\ & +\underbrace{c_k(h^2\left\|\mathbf{u}\right\|_{H^2}+h)\left\|\theta^{\mathbf{u}}\right\|_{L^2}}_{\text{by (3.63)}}+\underbrace{c_w h^2\left\|\mathbf{w}\right\|_{H^2}\left\|\theta^{\mathbf{w}}\right\|_{L^2}}_{\text{by (3.64)}} \\ & +\underbrace{c_w h^2\left\|\mathbf{w}\right\|_{H^2}\left\|\theta^{\mathbf{u}}\right\|_{L^2}}_{\text{by (3.64)}}+\underbrace{c_\psi h\left\|\mathbf{u}\right\|_{H^2}\left\|\nabla_{\Gamma_h^0}\theta^{\mathbf{u}}\right\|_{L^2}}_{\text{by (3.62)}}.\end{aligned}$$

Rearranging terms gives

$$\begin{aligned}\frac{d}{dt}m_h(\theta^{\mathbf{u}},\theta^{\mathbf{u}})+\left\|\theta^{\mathbf{w}}\right\|_{L^2}^2 \leq & C\left\|\theta^{\mathbf{u}}\right\|_{L^2}^2+c_\psi h\left\|\mathbf{u}\right\|_{H^2}\left\|\theta^{\mathbf{u}}\right\|_{H^1}+c_w h^2\left\|\mathbf{w}\right\|_{H^2}\left\|\theta^{\mathbf{w}}\right\|_{L^2} \\ & +c\left(h^2\left\|\partial_t\mathbf{u}\right\|_{H^2}+h^2\left\|\mathbf{u}\right\|_{H^2}+h+h^2\left\|\mathbf{w}\right\|_{H^2}\right)\left\|\theta^{\mathbf{u}}\right\|_{L^2}.\end{aligned}$$

We can suppress the H^1 norm of $\theta^{\mathbf{u}}$ using the same trick with (3.65) to suppress the dependence on $\nabla_{\Gamma^0}\theta^{\mathbf{u}}$. The right hand side can be split into dependence on θ and h only using Young's inequality, into h terms and θ ones. $\theta^{\mathbf{w}}$ can be shifted to the left.

$$\begin{aligned}\frac{d}{dt}m_h(\theta^{\mathbf{u}},\theta^{\mathbf{u}})+\left\|\theta^{\mathbf{w}}\right\|_{L^2}^2 \leq & C\left\|\theta^{\mathbf{u}}\right\|_{L^2}^2+Ch^4\left(\left\|\partial_t\mathbf{u}\right\|_{H^2}^2+\left\|\mathbf{u}\right\|_{H^2}^2+\left\|\mathbf{w}\right\|_{H^2}^2\right) \\ & +ch^2(\left\|\mathbf{u}\right\|_{H^2}^2+1).\end{aligned}$$

Again applying Gronwall inequality and integrating in time gives result. This is because the error $\theta^{\mathbf{u}}$ is zero at $t=0$.

□

3.10 Fully Discrete Version

To actually solve the problem we need to discretise in time. Problem 2.5.1 can be approximated in space as described by Problem 3.3.1, we now include a time stepping scheme and introduce a time parameter τ. For the time stepping we take all linear terms implicitly and nonlinear ones explicitly and we hence obtain a semi

implicit scheme. We then finish this chapter by providing a stability proof for the fully discrete problem defined next.

We split the time interval $[0,T]$ into M equal parts of size $\tau := T/M$. At each of these time steps we use the notation $t^m = \tau m$ for each $m = 0, 1, ..M-1$ and for a function we denote $\mathbf{f}^m(\cdot) = \mathbf{f}(\cdot, m\tau)$. For completeness we state the scheme here with all parameter terms included.

Problem 3.10.1. *For given constants $\lambda_{b,\psi}, \lambda_{l,\psi}, \lambda_{p,\psi}, l_0, u_B, k_L, u_{com}$ and given functions $\mathbf{U}_h^0, \mathbf{U}_h^c \in [H^1(\Gamma_h^0)]^3$, find $(\mathbf{U}_h^{m+1}, \mathbf{W}_h^{m+1}) \in [H^1(\Gamma_h^0)]^3 \times [H^1(\Gamma_h^0)]^3$ for all $m \in \{0, 1, ..., M-1\}$ such that for all $(\boldsymbol{\phi}_h, \boldsymbol{\chi}_h) \in [H^1(\Gamma_h^0)]^3 \times [H^1(\Gamma_h^0)]^3$*

$$\begin{aligned}
&\int_{\Gamma_h^0} \frac{1}{\tau}\mathbf{U}_h^{m+1}\boldsymbol{\phi}_h + \lambda_{b,\psi}\nabla_{\Gamma_h^0}\mathbf{W}_h^{m+1}\nabla_{\Gamma_h^0}\boldsymbol{\phi}_h \\
&\qquad + \nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}\nabla_{\Gamma_h^0}\boldsymbol{\phi}_h + k_{\epsilon_3}(|\mathbf{U}_h^m - \mathbf{U}_h^c|)\mathbf{U}_h^{m+1}\boldsymbol{\phi}_h \\
&\quad = \int_{\Gamma_h^0} \frac{1}{\tau}\mathbf{U}_h^m\boldsymbol{\phi}_h + 2x_0\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}\nabla_{\Gamma_h^0}\boldsymbol{\phi}_h + \frac{1}{|V_h^0|+\epsilon_2}\lambda_{p,\psi}\boldsymbol{\nu}_{\Gamma_h^0}\boldsymbol{\phi}_h \\
&\qquad + \int_{\Gamma_h^0} k_{\epsilon_3}(|\mathbf{U}_h^m - \mathbf{U}_h^c|)\left(\mathbf{U}_h^c + l_0\frac{\mathbf{U}_h^m - \mathbf{U}_h^c}{\sqrt{|\mathbf{U}_h^m - \mathbf{U}_h^c|^2+\epsilon_3}}\right)\boldsymbol{\phi}_h, \qquad (3.66)\\
0 &= \int_{\Gamma_h^0} \mathbf{W}_h^{m+1}\boldsymbol{\chi}_h - \nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}\nabla_{\Gamma_h^0}\boldsymbol{\chi}_h \qquad (3.67)
\end{aligned}$$

with $\epsilon_{1,2,3} > 0$, k_{ϵ_3} as in (2.18) and using $\boldsymbol{\nu}_{\Gamma_h^0}$ as in Definition 3.1.2. The discrete volume V_h^0 is computed as

$$V_h^0 := \frac{1}{3}\int_{\Gamma_h^0} \mathbf{U}_h^0\boldsymbol{\nu}_{\Gamma_h^0}d\sigma_h$$

We now outline the matrix vector formulation for this problem using the notation introduced in Section 3.5 when possible. The full scheme is solved by using the biconjugate gradient method. The fields are expressed in terms of the basis functions of $\mathbf{S}_h^3$: with degrees of freedom vector $\boldsymbol{\eta}^{\mathbf{u},m}, \boldsymbol{\eta}^{\mathbf{w},m} \in \mathbb{R}^{3N}$, that is split in the spatial directions $\beta = 1, 2, 3$, and discrete nodes $i = 1, .., N := dim(S_h)$, for each time step $m = 0, 1, ..., M-1$ together we obtain

$$\mathbf{U}_h^m(\mathbf{x}) := \sum_{\beta=1}^{3}\sum_{i=1}^{N}\eta_{(\beta-1)N+i}^{\mathbf{u},m}\phi_i(\mathbf{x})\mathbf{e}_\beta, \quad \mathbf{W}_h^m(\mathbf{x}) := \quad \sum_{\beta=1}^{3}\sum_{i=1}^{N}\eta_{(\beta-1)N+i}^{\mathbf{w},m}\phi_i(\mathbf{x})\mathbf{e}_\beta,$$

for $\mathbf{x} \in \Gamma_h^0$ where $\{\mathbf{e}_\beta\}_{\beta=1}^3$ denote the standard basis vectors of $\mathbb{R}^3$. We define

$\eta^c_{(\beta-1)N+i} := (\boldsymbol{\phi}_{\beta,i}(\mathbf{x}), \mathbf{u}_c(\mathbf{x}))$ to then obtain

$$\mathbf{U}^c_h(\mathbf{x}) := \sum_{\beta=1}^{3}\sum_{i=1}^{N} \eta^c_{(\beta-1)N+i}\phi_i(\mathbf{x})\mathbf{e}_\beta,$$

and for the basic matrices we reuse (3.30). We now define the implicit part of the linker term as

$$M^k_{(\beta-1)N+i,(\gamma-1)N+j}(\boldsymbol{\eta}^{\mathbf{u},m}) = \int_{\Gamma^0_h} \delta_{\gamma,\beta}\mathbf{k}_{\epsilon_3}\left(\sqrt{\mathbf{U}^m_h(\mathbf{x}) - \mathbf{U}^c_h(\mathbf{x})}\right)\phi_i(\mathbf{x})\phi_j(\mathbf{x})d\sigma_h.$$

We now look to define the explicit terms. For the nonlinear gradient term we obtain

$$D(\boldsymbol{\eta}^{\mathbf{u},m})_{(\gamma-1)N+j} := \int_{\Gamma^0_h} 2\delta_{\gamma,\beta}x_0\frac{\nabla_{\Gamma^0_h}\phi_i(\mathbf{x})\cdot\nabla_{\Gamma^0_h}\phi_j(\mathbf{x})}{\sqrt{|\nabla_{\Gamma^0_h}U^m_h(\mathbf{x})|^2+\epsilon_1}}d\sigma_h.$$

For the explicit linker we use

$$\eta^l_{(\beta-1)N+i} := \frac{(\eta^{\mathbf{u},m}_{(\beta-1)N+i} - \eta^c_{(\beta-1)N+i})}{\sqrt{\sum_{\alpha=1}^{3}\left(\eta^{\mathbf{u},m}_{(\alpha-1)N+i} - \eta^c_{(\alpha-1)N+i}\right)^2+\epsilon_3}}.$$

For the pressure we use

$$P_{(\gamma-1)N+j} := \int_{\Gamma^0_h}\frac{\lambda_{p,\psi}\phi_j(\mathbf{x})}{|V^0_h|+\epsilon_2}(\boldsymbol{\nu}_{\Gamma^0_h}(\mathbf{x})\cdot\mathbf{e}_\gamma)d\sigma_h.$$

Here $\boldsymbol{\nu}_{\Gamma^0_h}$ is as outlined in definition 3.1.2. The matrix vector formulation of scheme 3.10.1 is then

$$\begin{aligned}\left(\frac{\omega}{\tau}\mathbf{M}+\mathbf{S}+\mathbf{M}_k(\boldsymbol{\eta}^{\mathbf{u},m})\right)\boldsymbol{\eta}^{\mathbf{u},m+1} + \lambda_{b,\psi}\mathbf{S}\boldsymbol{\eta}^{\mathbf{w},m+1} =& \frac{\omega}{\tau}\mathbf{M}\boldsymbol{\eta}^{\mathbf{u},m} + D(\boldsymbol{\eta}^{\mathbf{u},m}) \\ &+ \mathbf{M}_k(\boldsymbol{\eta}^{\mathbf{u},m})(\boldsymbol{\eta}^{c,m}+l_0\boldsymbol{\eta}^l)+\mathbf{P}.\end{aligned} \tag{3.68}$$

This means we need to solve the following system for each $\gamma = 1,2,3$ and $i = 1,...,N$ at each time step $m = 0,1,2,...,M-1$.

3.10.1 Stability of Fully Discrete Scheme 3.10.1

Lemma 3.10.1. *(Stability)The solution* $\mathbf{U}_h^m$ *of the fully discrete scheme satisfies the following stability estimate*

$$\left\|\mathbf{U}_h^{m+1}\right\|^2 + \tau c_0 \sum_{m=0}^{M-1} \left(\left\|\mathbf{W}_h^{m+1}\right\|^2 + \left\|\nabla_{\Gamma_h^0} \mathbf{U}_h^{m+1}\right\|^2 \right) \leq C_1 \tau (M-1)(\tau+1) + C_2 \left\|\mathbf{U}_h^0\right\|^2 .$$

Proof. We do not state the precise constants in the proof and assume that they are equal to one (for improved readability). First we show some identities that will be used later on in the proof.

$$\begin{aligned}
&m_h(\mathbf{U}_h^{m+1} - \mathbf{U}_h^m, \mathbf{U}_h^m) \\
=&\frac{1}{2} m_h(\mathbf{U}_h^{m+1} - \mathbf{U}_h^m, \mathbf{U}_h^{m+1} + \mathbf{U}_h^m) - \frac{1}{2} m_h(\mathbf{U}_h^{m+1} - \mathbf{U}_h^m, \mathbf{U}_h^{m+1} - \mathbf{U}_h^m) \\
=&\frac{1}{2} \left\|\mathbf{U}_h^{m+1}\right\|^2 - \frac{1}{2} \left\|\mathbf{U}_h^m\right\|^2 \underbrace{-\frac{1}{2} m_h(\mathbf{U}_h^m, \mathbf{U}_h^{m+1}) + \frac{1}{2} m_h(\mathbf{U}_h^{m+1}, \mathbf{U}_h^m)}_{=0} - \frac{1}{2} \left\|\mathbf{U}_h^{m+1} - \mathbf{U}_h^m\right\|^2 ,
\end{aligned}$$

which using the symmetry of $m_h(\cdot,\cdot)$ gives the identity

$$m_h(\mathbf{U}_h^{m+1} - \mathbf{U}_h^m, \mathbf{U}_h^m) = \frac{1}{2} \left\|\mathbf{U}_h^{m+1}\right\|^2 - \frac{1}{2} \left\|\mathbf{U}_h^m\right\|^2 - \frac{1}{2} \left\|\mathbf{U}_h^{m+1} - \mathbf{U}_h^m\right\|^2 . \tag{3.69}$$

We also define for general vector valued functions $\mathbf{f}_1, \mathbf{f}_2 \in [H^1(\Gamma_h^0)]^3$ and scalar function $f_3 \in H^1(\Gamma_h^0)$

$$\begin{aligned}
m_h(f_3^2(\mathbf{f}_1 - \mathbf{f}_2), \mathbf{f}_1) =& \frac{1}{2} m_h(f_3^2(\mathbf{f}_1 - \mathbf{f}_2), \mathbf{f}_1 - \mathbf{f}_2) + \frac{1}{2} m_h(f_3^2(\mathbf{f}_1 - \mathbf{f}_2), \mathbf{f}_1 + \mathbf{f}_2) \\
=& \frac{1}{2} \underbrace{\left\|f_3(\mathbf{f}_1 - \mathbf{f}_2)\right\|^2}_{\geq 0} + \frac{1}{2} \left\|f_3 \mathbf{f}_1\right\|^2 - \frac{1}{2} \left\|f_3 \mathbf{f}_2\right\|^2 \underbrace{- m_h(f_3^2 \mathbf{f}_2, \mathbf{f}_1) + m_h(f_3^2 \mathbf{f}_1, \mathbf{f}_2)}_{=0},
\end{aligned}$$

which gives the identity

$$m_h(f_3^2(\mathbf{f}_1 - \mathbf{f}_2), \mathbf{f}_1) \geq \frac{1}{2} \left\|f_3 \mathbf{f}_1\right\|^2 - \frac{1}{2} \left\|f_3 \mathbf{f}_2\right\|^2 . \tag{3.70}$$

To enhance readability we introduce the function $\mathbf{K}_h$

$$\mathbf{K}_h(\mathbf{A}, \mathbf{B}) := k_{\epsilon_3}(|\mathbf{B} - \mathbf{U}_h^c|) \left(\mathbf{A} - \mathbf{U}_h^c + \frac{\mathbf{B} - \mathbf{U}_h^c}{\sqrt{|\mathbf{B} - \mathbf{U}_h^c|^2 + \epsilon_3}} \right).$$

We can now write the Problem 3.10.1 in a reduced form, incorporating dropped

parameters and grouped term $\mathbf{K}$

$$\begin{aligned}
&\frac{1}{\tau}m_h(\mathbf{U}_h^{m+1}-\mathbf{U}_h^m,\boldsymbol{\phi}_h)+s_h(\mathbf{W}_h^{m+1},\boldsymbol{\phi}_h)\\
&+s_h(\mathbf{U}_h^{m+1},\boldsymbol{\phi}_h)+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\boldsymbol{\phi}_h)=\int_{\Gamma_h^0}2\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}\boldsymbol{\phi}_h}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h\\
&\qquad\qquad+m_h\left(\frac{1}{|V_h^0|+\epsilon_2}\boldsymbol{\nu}_{\Gamma_h^0},\boldsymbol{\phi}_h\right). \quad (3.71)
\end{aligned}$$

Now first take $\boldsymbol{\phi}_h=\mathbf{U}_h^{m+1}-\mathbf{U}_h^m$ in (3.71) to get

$$\begin{aligned}
&\frac{1}{\tau}\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|^2+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)\\
&\qquad+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)\\
&+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)=\int_{\Gamma_h^0}2\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}(\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h\\
&\qquad\qquad+m_h\left(\frac{1}{|V_h^0|+\epsilon_2}\boldsymbol{\nu}_{\Gamma_h^0},\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right).
\end{aligned}$$

We can now apply Cauchy Schwarz then Young with a weighting τ to the pressure term

$$\begin{aligned}
m_h\left(\frac{1}{|V_h^0|+\epsilon_2}\boldsymbol{\nu}_{\Gamma_h^0},\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right)\leq&\left\|\frac{1}{(|V_h^0|+\epsilon_2)}\boldsymbol{\nu}_{\Gamma_h^0}\right\|\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|\\
&\frac{\tau}{2(|V_h^0|+\epsilon_2)^2}\left\|\boldsymbol{\nu}_{\Gamma_h^0}\right\|^2+\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|^2,
\end{aligned}$$

this allows us to suppress the $\mathbf{U}_h^{m+1}-\mathbf{U}_h^m$ dependence.

$$\begin{aligned}
&\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|^2+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)\\
&\qquad+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)\\
&+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)\leq\int_{\Gamma_h^0}2\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}(\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h\\
&\qquad\qquad+\frac{|\Gamma_h^0|^2\tau}{2(|V_h^0|+\epsilon_2)^2}.
\end{aligned}$$

This inequality can then be rewritten with terms transferred to the right hand side

$$\begin{aligned}\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|^2 \leq & s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^m-\mathbf{U}_h^{m+1})+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m-\mathbf{U}_h^{m+1}) \\ & +\int_{\Gamma_h^0}\frac{2\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}(\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon}}d\sigma_h \\ & +m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^m-\mathbf{U}_h^{m+1})+\frac{|\Gamma_h^0|^2\tau}{2(|V_h^0|+\epsilon)^2}. \quad (3.72)\end{aligned}$$

We now return to (3.71) taking $\boldsymbol{\phi}_h=\mathbf{U}_h^m$

$$\begin{aligned}&\frac{1}{\tau}m_h(\mathbf{U}_h^{m+1}-\mathbf{U}_h^m,\mathbf{U}_h^m)+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^m) \\ &+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m)+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^m)=\int_{\Gamma_h^0}\frac{2\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}\mathbf{U}_h^m)}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon}}d\sigma_h \\ &\qquad +m_h\left(\frac{1}{|V_h^0|+\epsilon}\boldsymbol{\nu}_{\Gamma_h^0},\mathbf{U}_h^m\right).\end{aligned}$$

Now we use our pre proved identities (3.69) and (3.70). Using (3.69) we get

$$\begin{aligned}&\frac{1}{\tau}\left(\frac{1}{2}\left\|\mathbf{U}_h^{m+1}\right\|^2-\frac{1}{2}\left\|\mathbf{U}_h^m\right\|^2-\frac{1}{2}\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|^2\right) \\ &\qquad +s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^m)+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m) \\ &\qquad +m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^m)\leq\int_{\Gamma_h^0}\frac{2\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}\mathbf{U}_h^m}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon}}d\sigma_h \\ &\qquad +m_h\left(\frac{1}{|V_h^0|+\epsilon}\boldsymbol{\nu}_{\Gamma_h^0},\mathbf{U}_h^m\right).\end{aligned}$$

We rearranges to

$$\begin{aligned}&\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}\right\|^2-\frac{1}{2\tau}\left\|\mathbf{U}_h^m\right\|^2+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^m) \\ &+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m)+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^m)\leq \quad \int_{\Gamma_h^0}2\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}\mathbf{U}_h^m}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h \\ &\qquad +\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}-\mathbf{U}_h^m\right\|^2 \\ &\qquad +\frac{|\Gamma_h^0|}{|V_h^0|+\epsilon_2}\left\|\mathbf{U}_h^m\right\|.\end{aligned}$$

This allows us to use (3.72)

$$\begin{aligned}
&\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}\right\|^2-\frac{1}{2\tau}\left\|\mathbf{U}_h^{m}\right\|^2+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^m)\\
+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m)+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^m)\leq&\int_{\Gamma_h^0}\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m:\nabla_{\Gamma_h^0}\mathbf{U}_h^m}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h\\
&+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^m-\mathbf{U}_h^{m+1})\\
&+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m-\mathbf{U}_h^{m+1})\\
&+\int_{\Gamma_h^0}2\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m\nabla_{\Gamma_h^0}(\mathbf{U}_h^{m+1}-\mathbf{U}_h^m)}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h\\
&+\frac{|\Gamma_h^0|}{|V_h^0|+\epsilon_2}\left\|\mathbf{U}_h^m\right\|+\frac{|\Gamma_h^0|^2\tau}{2(|V_h^0|+\epsilon_2)^2}\\
&+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^m-\mathbf{U}_h^{m+1}).
\end{aligned}$$

By cancelling terms this reduces to

$$\begin{aligned}
&\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}\right\|^2-\frac{1}{2\tau}\left\|\mathbf{U}_h^{m}\right\|^2\\
+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^{m+1})+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^{m+1})&\\
+m_h(\mathbf{K}_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^m),\mathbf{U}_h^{m+1})\leq&\int_{\Gamma_h^0}2\frac{\nabla_{\Gamma_h^0}\mathbf{U}_h^m\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h\\
&+\frac{|\Gamma_h^0|}{|V_h^0|+\epsilon_2}\left\|\mathbf{U}_h^m\right\|+\frac{|\Gamma_h^0|^2\tau}{2(|V_h^0|+\epsilon_2)^2}.
\end{aligned}$$

We now apply (3.70) for the components of the linker term taking

$$\mathbf{f}_1=\mathbf{U}_h^{m+1},\qquad \mathbf{f}_2=\mathbf{U}_h^c+\frac{\mathbf{U}_h^m-\mathbf{U}_h^c}{\sqrt{|\mathbf{U}_h^m-\mathbf{U}_h^c|^2+\epsilon_3}},\qquad f_3^2=k_{\epsilon_3}(|\mathbf{U}_h^m-\mathbf{U}_h^c|).$$

This allows us to rewrite the linker term.

$$\begin{aligned}
&\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}\right\|^2-\frac{1}{2\tau}\left\|\mathbf{U}_h^{m}\right\|^2+s_h(\mathbf{W}^{m+1},\mathbf{U}^{m+1})+s_h(\mathbf{U}_h^{m+1},\mathbf{U}_h^{m+1})\\
&+\underbrace{\frac{1}{2}\left\|\sqrt{k_{\epsilon_3}(|\mathbf{U}_h^m-\mathbf{U}_h^c|)}\mathbf{U}_h^{m+1}\right\|^2}_{\geq 0}\\
\leq&\int_{\Gamma_h^0}\frac{2\nabla_{\Gamma_h^0}\mathbf{U}_h^m\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon_1}}d\sigma_h+\frac{|\Gamma_h^0|}{|V_h^0|+\epsilon}\left\|\mathbf{U}_h^m\right\|+\frac{|\Gamma_h^0|^2\tau}{2(|V_h^0|+\epsilon_2)^2}\\
&+\frac{1}{2}\left\|\sqrt{k_{\epsilon_3}(|\mathbf{U}_h^m-\mathbf{U}_h^c|)}\left(\mathbf{U}_h^c+\frac{\mathbf{U}_h^m-\mathbf{U}_h^c}{\sqrt{|\mathbf{U}_h^m-\mathbf{U}_h^c|^2+\epsilon_3}}\right)\right\|^2.
\end{aligned}$$

The right hand side of the linker term can be written in terms of the indicator function $\sqrt{k_{\epsilon_3}(|\mathbf{U}_h^m-\mathbf{U}_h^c|)}$ and initial data. The left hand side's linker component can be dropped as the term is positive. The nonlinear tension term on the right hand side can be bounded and then be split up by Young (with suitable weighting) to suppress $\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}$ on the left hand side.

$$\int_{\Gamma_h^0}\frac{2\nabla_{\Gamma_h^0}\mathbf{U}_h^m\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}}{\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon}}d\sigma_h\leq 2\frac{\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h^m\right\|}{\left\|\sqrt{|\nabla_{\Gamma_h^0}\mathbf{U}_h^m|^2+\epsilon}\right\|}\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}\right\|\leq 2|\Gamma_h^0|\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}\right\|.$$

Putting this all together we get

$$\begin{aligned}
&\frac{1}{2\tau}\left\|\mathbf{U}_h^{m+1}\right\|^2-\frac{1}{2\tau}\left\|\mathbf{U}_h^{m}\right\|^2\\
+s_h(\mathbf{W}_h^{m+1},\mathbf{U}_h^{m+1})+\frac{1}{2}\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}\right\|\leq&4|\Gamma_h^0|+\frac{|\Gamma_h^0|}{|V_h^0|+\epsilon}\left\|\mathbf{U}_h^m\right\|+\frac{|\Gamma_h^0|^2\tau}{2(|V_h^0|+\epsilon)^2}\\
&+C\left\|\sqrt{k(|\mathbf{U}_h^m-\mathbf{U}_h^c|)}\right\|^2.
\end{aligned}$$

Finally we can take $\boldsymbol{\chi}_h=\mathbf{W}^{m+1}$ in (3.67) which allows us to substitute out the left hand side term of $\mathbf{W}_h^{m+1}$.

$$\left\|\mathbf{U}_h^{m+1}\right\|^2+\tau\left\|\mathbf{W}_h^{m+1}\right\|^2+\tau\left\|\nabla_{\Gamma_h^0}\mathbf{U}_h^{m+1}\right\|^2\leq C_1\tau(\tau+1)+C_2\left\|\mathbf{U}_h^m\right\|^2$$

We can then apply a discrete Gronwall inequality (Lemma A.0.3) to get the result. □

Remark 3.10.1. *An alternative for the scheme is to use a mixed term for the non-*

linear gradient.

$$\left(\nabla_{\Gamma^0}\mathbf{u}^{m+1} - \frac{2x_0\nabla_{\Gamma^0}\mathbf{u}^{m+1}}{\sqrt{|\nabla_{\Gamma^0}\mathbf{u}^m|^2+\epsilon}}\right)\nabla_{\Gamma^0}\boldsymbol{\varphi}.$$

While this does lead to a stable scheme, the right hand side now contains a dependence on ϵ^{-1} for bounding the explicit part of the non linearity. As we take ϵ small this destroys much of the control over a solution.

Remark 3.10.2. *Alternatively the linker term can be taken completely explicitly in time. The proof can accommodate this by handling it the same way as the pressure. This would be the recommended approach if users want to consider advanced linker formulations.*

In this chapter we started with Problem 2.5.1 that contains operator splitting and generalized to a more abstract set of equations. We were able to show weak well posedness by using a semi discrete surface finite element scheme. This proof used the existence in the semi discrete case and bounds obtained on the solution $(\mathbf{U}_h, \mathbf{W}_h)$ to prove results about the limit. After the well posedness proof, by assuming some regularity we were able to show some error bounds for the semi discrete scheme.

The second part of the chapter makes the model fully discrete in time by adding a time stepping scheme (see Problem 3.10.1). Here we chose a semi implicit scheme that takes all linear terms implicitly and nonlinear ones explicitly. We show Problem 3.10.1 is stable in Lemma 3.10.1. Now that we have a fully discrete formulation, in the next chapter we implement the scheme and try to show the convergence rate experimentally. Afterwards we show examples of the model predicting bleb sites for various initial surfaces.

Chapter 4

Numerical Simulations

In this chapter we cover various simulations carried out using our model. First we look to validate the model as a correct implementation by showing experimental orders of convergence in two example cases. Afterwards, we show how well the model matches up with its intended purpose. In Section 1.2 we defined what blebbing involves and discussed ideas of how they form. We will show that the model reproduces the discussed cellular behaviour for parameters taking realistic values. In particular, the connection between linkers, tension, and pressure can now be investigated.

The model from Problem 3.10.1 was implemented in DUNE-FEM [12] using linear surface finite elements. To find suitable examples we include a forcing term given by additional function $\mathbf{f}(\mathbf{x}, t)$ when studying examples for the experimental order of convergence. This means we do not need to find analytical solutions to Problem 2.5.1 as it is stated, which by being fourth order may be difficult. By taking Problem 2.5.1 with an additional term of the form $m(\mathbf{f}, \boldsymbol{\phi})$ we can then, for our choice of $\mathbf{u}$, define $\mathbf{f}$ such that

$$\begin{aligned}\mathbf{f} :=& \partial_t \mathbf{u} - \lambda_{b,\psi} \Delta_{\Gamma^0}^2 \mathbf{u} - \nabla_{\Gamma^0} \cdot (\nabla_{\Gamma^0} \mathbf{u} - 2x_0 \frac{\nabla_{\Gamma^0} \mathbf{u}}{\sqrt{|\nabla_{\Gamma^0} \mathbf{u}|^2 + \epsilon_1}}) \\ & - k_{\epsilon_3}(|\mathbf{u} - \mathbf{u}_c|) \left(\mathbf{u} - \mathbf{u}_c - l_0 \frac{\mathbf{u} - \mathbf{u}_c}{|\mathbf{u} - \mathbf{u}_c| + \epsilon_3} \right) - \frac{\lambda_{p,\psi}}{V^0 + \epsilon_2} \boldsymbol{\nu}_{\Gamma^0}.\end{aligned} \tag{4.1}$$

The function $\mathbf{u}$ being given by our choice allows us to generate examples that we can then test the error and compute the rate of convergence. In practice, (4.1) was computed with the aid of Mathematica [40], this is because calculating (4.1) with both fourth order terms and nonlinear gradient is non trivial. The fully discrete Problem 3.10.1 is then also adapted to include a similar forcing term. In the function k_{ϵ_3} defined in (2.18), we let $k_L = 0$ for all examples measuring the convergence.

The function is removed as its size rapidly changes over distance $\epsilon_3 > 0$ around u_{com} which creates a phase changes and could skew the numerical results if $\epsilon_3 < h$ when this transition will not be computed accurately. As we are not concerned by behaviour as $\epsilon_3 \to 0$ we simply remove the term. The parameter k_L will feature for examples of bleb simulation in Section 4.3.1. For parameters we take $\epsilon_{1,2} = 10^{-10}$ and 1 for $\lambda_{p,\psi}, l_0, u_B, \lambda_{l,\psi}, \lambda_{b,\psi}$ unless stated otherwise. The cortex $\mathbf{u}_c$ and time parameters are stated for each example.

For all examples, we solve Problem 3.10.1 using the bi conjugate gradient method [37] as the solver needs to invert a nonsymmetric linear system. For the error we consider the L^2 error at the final time step which for $\mathbf{u}$ is

$$\left\| \mathbf{u}^{-l}(\cdot, T) - \mathbf{U}_h^M(\cdot) \right\|_{L^2(\Gamma_h^0)}, \tag{4.2}$$

and also the H^1 error

$$\left\| \nabla_{\Gamma_h^0} \mathbf{u}^{-l}(\cdot, T) - \nabla_{\Gamma_h^0} \mathbf{U}_h^M(\cdot) \right\|_{L^2(\Gamma_h^0)}. \tag{4.3}$$

We consider $\mathbf{w}$ error in the same way. To analyse these it is best to compute the order of convergence.

Definition 4.0.1. *The experimental order of convergence (eoc) for a sequence of triangulations $\{\mathfrak{T}_i\}_{i=1}^N$ with error of size $\{E_i\}_{i=1}^N$ and given mesh size h_i of triangulation $\{\mathfrak{T}_i\}_{i=1}^N$ is defined by*

$$(eoc)_i := \frac{\log(E_i/E_{i-1})}{\log(h_i/h_{i-1})}. \tag{4.4}$$

This gives us a measure of the convergence rates which can then be compared to those of standard linear finite elements.

We do not carry out any temporal convergence test as the semi implicit time stepping scheme in (Problem 3.10.1) is relatively simple. In each example, we will state and motivate the choice of $\mathbf{u}$, $\mathbf{u}_c$ and Γ^0. To define a surface we will sometimes use a level set function $\Phi(\mathbf{x})$. This is a scalar-valued function were $\Phi(\mathbf{x}) = 0$ if and only if $\mathbf{x} \in \Gamma^0$. All post processing of data to show images is done using PARAVIEW [3].

4.1 Example: Shrinking Ball

For the first example, we take a decreasing solutions over the unit ball. The surface is given by the zero level set of the function

$$\Gamma^0 := \{\mathbf{x} \in \mathbb{R}^3 : 0 = |\mathbf{x}| - 1\},$$

with cortex defined by

$$\mathbf{u}_c(\mathbf{x}) := 0.6\frac{\mathbf{x}}{|\mathbf{x}|}, \qquad \mathbf{x} \in \Gamma^0. \tag{4.5}$$

The parameters used here are $l_0 = 0.2$ and $u_B = 0.6$. We look for the error at final time $T = 0.2$. For time step we initially use $\tau = 0.1$ and after each refinement divide it by 4.

For this test case we take the exact solution $\mathbf{u} : S^2 \times [0, 0.2] \to \mathbb{R}^3$,

$$\mathbf{u}(\mathbf{x}, t) = (1 - t)\mathbf{x}. \tag{4.6}$$

We present the resulting convergence rates in Tables 4.1 and 4.2 which show quadratic order of convergence for the L^2 errors in $\mathbf{U}_h$ and $\mathbf{W}_h$. The expectation is for linear convergence because the error for approximating $\boldsymbol{\nu}_{\Gamma^0}$ is $O(h)$ (by (3.6)). However as the surface is a sphere, the symmetry may give faster convergence. We see linear convergence $O(h)$ as expected for the gradients $\nabla_{\Gamma_h^0}\mathbf{W}_h$ and $\nabla_{\Gamma_h^0}\mathbf{U}_h$.

4.2 Example: Dziuk Surface

Now we consider the model over a more complex surface that is given by the level set

$$\Gamma^0 := \{\mathbf{x} : (x_1 - x_3^2)^2 + x_2^2 + x_3^2 - 1 = 0\}. \tag{4.7}$$

The surface (4.7) has a projection as it is well defined locally around the surface and it is smooth and closed. In [11] they claim however that no explicit formula exists for the projection so we adopt an ad-hoc method used therein. We summarise the

elements	h	$\left\|u^{-l}(\cdot, M\tau) - U_h^M\right\|_{L^2(\Gamma_h^0)}$	eoc
12	1.63299	1.16958	-
48	1	0.50849	1.698464489
192	0.632456	0.16544	2.450914007
768	0.342997	0.04460	2.142295178
3072	0.175412	0.0113882	2.035768019
12288	0.0882162	0.00286	2.008622491
elements	h	$\left\|\nabla_{\Gamma_h^0}(u^{-l}(\cdot, M\tau) - U_h^M)\right\|_{L^2(\Gamma_h^0)}$	eoc
12	1.63299	1.94022	-
48	1	1.13826	1.087452667
192	0.632456	0.63834	1.26244044
768	0.342997	0.32627	1.096840145
3072	0.175412	0.16463	1.020079316
12288	0.0882162	0.08247	1.005744023

Table 4.1: Experimental order of convergence for the Example 4.1 looking at the convergence in $\mathbf{u}$

algorithm for projection in the next subsection. The resulting discrete surface now projects within some tolerance. These are chosen sufficiently small such that it does not affect the numerical tests.

4.2.1 Surface Approximation

When a mesh is refined new points are created and in the case of a curved domain these points need to be projected. If this doesn't happen the mesh does not improve the approximation of the surface. As level set (4.7) has no projection that we know of the surface must be approximated another way. Instead, an algorithm [11] approximates the projection of points. See Figure 4.2 for an example mesh after three refinements.

Assume that the surface is given as the zero level set of some function $\Phi(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}$ as in (4.7) with positive values outside Γ^0 and negative inside. This means $\Phi(\mathbf{x}) = 0$ hold if and only if $\mathbf{x} \in \Gamma^0$ and also require that the gradient of the level set is equal to the normal $\boldsymbol{\nu}_{\Gamma^0}(\mathbf{x}) = \nabla\Phi(\mathbf{x})/|\nabla\Phi(\mathbf{x})|$ for all $\mathbf{x} \in \Gamma^0$. By specifying the level set to be negative inside the surface we chose the orientation. Using this level set function the algorithm runs until (4.8) reaches a suitable tolerance. By minimizing deviations from 0 in the first part of the condition we find points that are closer to the surface. A second condition is added to (4.8) that optimize for points in the normal direction. Note this is only an approximation of the projection

elements	h	$\left\|w^{-l}(\cdot, M\tau) - W_h^M\right\|_{L^2(\Gamma_h^0)}$	eoc
12	1.63299	0.662884	-
48	1	0.279301	1.762413527
192	0.632456	0.0900051	2.471760343
768	0.342997	0.025352	2.070651396
3072	0.175412	0.00656036	2.015872319
12288	0.0882162	0.0016582	2.000900647
elements	h	$\left\|\nabla_{\Gamma_h^0}(w^{-l}(\cdot, M\tau) - W_h^M)\right\|_{L^2(\Gamma_h^0)}$	eoc
12	1.63299	3.83771	-
48	1	2.25053	1.088287683
192	0.632456	1.27753	1.235935523
768	0.342997	0.638343	1.133879025
3072	0.175412	0.3293	0.9870595803
12288	0.0882162	0.164938	1.005896107

Table 4.2: Experimental order of convergence for Example 4.1 looking at the convergence in $\mathbf{w}$.

$\mathbf{p}(\mathbf{x})$ seen in (3.1), while the algorithm converges to a point on Γ^0 it does not generally converge to unique point $\mathbf{p}(\mathbf{x})$.

In practise it was necessary to precondition $(\Phi(\mathbf{x}) + 1)^{-1}$ to better move points towards the centre of the element. This means instead of using the middle point of the triangle $\mathbf{x}_0 = \mathbf{x}$, we use $\mathbf{x}_0 = (\Phi(\mathbf{x}) + 1)^{-1}\mathbf{x}$. This gives points which are not too close to the edges and therefore not reducing the size of h by half, Figure 4.1 is an illustrative example.

Algorithm 4.2.1. *In this algorithm, we take* $\mathbf{x}$*, the new mesh point made by partitioning an element and return a new point* $\tilde{\mathbf{p}}(\mathbf{x})$ *which is an approximation of a projection onto the surface. We assume that any initial guess of* $\mathbf{x}$ *is close to the target surface.*

1. *Set* $\mathbf{x}_0 := \mathbf{x}/(\Phi(\mathbf{x}) + 1)$*. Initialise* $k := 0$

2. *Check if the following holds:*

$$\sqrt{\left(\frac{\Phi(\mathbf{x}_k)^2}{|\nabla\Phi(\mathbf{x}_k)|^2} + \left|\frac{\mathbf{x}_k - \mathbf{x}_0}{|\mathbf{x}_k - \mathbf{x}_0|} - \frac{\nabla\Phi(\mathbf{x}_k)}{|\nabla\Phi(\mathbf{x}_k)|}\right|\right)} < tol. \tag{4.8}$$

If true then finish and return $\tilde{\mathbf{p}}(\mathbf{x}) := \mathbf{x}_k$*, otherwise move to step 3.*

3. Compute the following

$$\hat{\mathbf{x}}_k := \mathbf{x}_k - \Phi(\mathbf{x}_k)\frac{\nabla\Phi(\mathbf{x}_k)}{|\nabla\Phi(\mathbf{x}_k)|^2}, \qquad d_k := sign(\Phi(\mathbf{x}_0))|\hat{\mathbf{x}}_k - \mathbf{x}_0|.$$

4. Update the guess of $\mathbf{x}$*,*

$$\mathbf{x}_{k+1} := \mathbf{x}_0 - d_k\frac{\nabla\Phi(\hat{\mathbf{x}_k})}{|\nabla\Phi(\hat{\mathbf{x}_k})|}.$$

5. Return to step 2 and set $k := k + 1$*.*

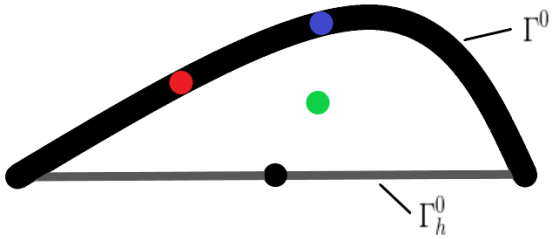


Figure 4.1: When refining a mesh the new black point is generated and normally projected to the surface Γ_h^0. As the projection is not known the algorithm is used. The original Algorithm in [11] maps the black spot to the red one. This does not produce a good mesh as seen in the bottom of Figure 4.2. In turn, our algorithm instead computes the blue dot thanks to the precondition that moves the starting value to the green point. For the Dziuk surfaces, this seems to give a better mesh, see top images in Figure 4.2.

4.2.2 Dziuk Surface Test Case

For this surface we take the solution

$$\mathbf{u}(\mathbf{x}, t) := (1 + t)\mathbf{x}. \tag{4.9}$$

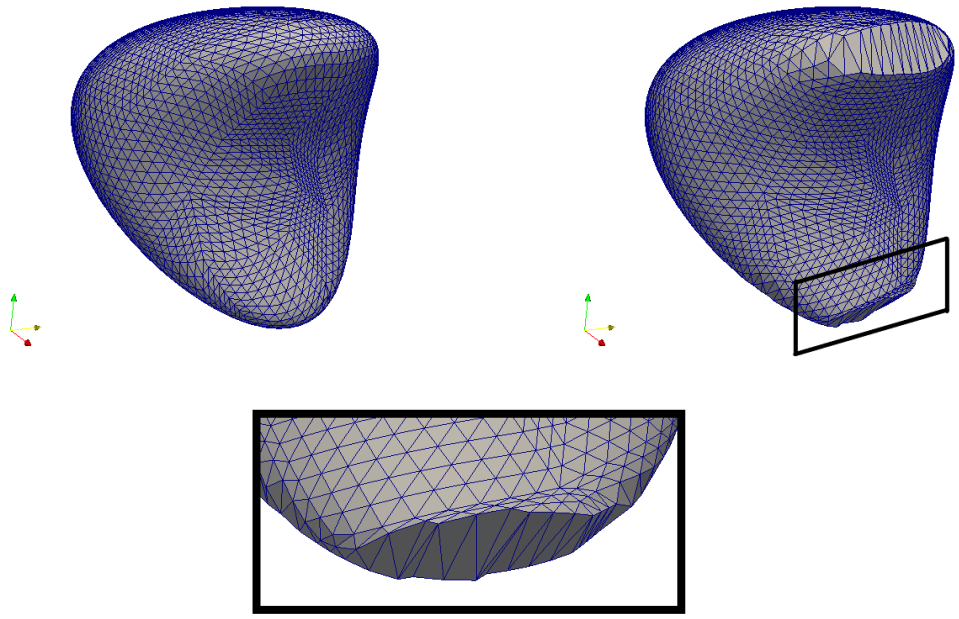


Figure 4.2: On the top left is a mesh used for (4.7) after three refinements. On the top right is a picture of the same number of refinements but without the preconditioning discussed in Algorithm 4.2.1, instead using the algorithm as it appears in [11]. Notice the mesh is not refined correctly at the poles. This create large elements with very small triangles built up adjacently where the algorithm returns a point close to the triangles edge. A magnified image of the black box is provided in the bottom image.

The numerical parameters for this example are $\tau = 0.00001$ and for the linker terms $l_0 = 0.5$ and $u_B = 1$ with a sphere as the cortex with radius 0.5. We reduced the bending to $\lambda_{b,\psi} = 0.1$ as the solver seemed to perform better when the system matrix defined by (3.68) was more diagonally dominated. Larger values of $\lambda_{b,\psi}$ required iterations of the solver which greatly slowed the solve time of the system. Using a lower value of $\lambda_{b,\psi}$ was sufficient to allow the solver to work without requiring preconditioning. This change seemed to have little effect on the resulting error values. We set the cortex function to be

$$\mathbf{u}_c(\mathbf{x}) := 0.5\frac{\mathbf{x}}{|\mathbf{x}|}, \qquad \mathbf{x} \in \Gamma^0. \tag{4.10}$$

The simulation ran until $T = 0.01$. The experimental order of convergence are given in Tables 4.3 and 4.4. The volume $|V^0|$ from Problem 2.5.1 is approximated by a heavily refined approximation of the surface (6 refinements).

In Table 4.3 we do not see a linear convergence rate of the L^2 error. As the normal $\boldsymbol{\nu}_{\Gamma^0}$ (Lemma 3.1.2) converges linearly we might expect the scheme to converge at a similar rate. This may suggest we have some level of superconvergence in the scheme, in reality, the error produced by the normal is much smaller than the other terms for this mesh. To test this would require several more refinements before we potentially see linear convergence rate. One possible influence is the $\nabla_{\Gamma^0}\mathbf{w}$ which is approximated poorly (see Table 4.4), may dominate the convergence. This contribution comes mainly from the poles which our projection seems to poorly approximate. If we decrease $\lambda_{b,\psi}$ the rate does not change. Another reason could be the chosen $\mathbf{u}$ is too simple. For example our $\mathbf{u}$ (4.9) gives $|\nabla_{\Gamma^0}\mathbf{u}|^2 = 2(1+t)$ which is not spatially dependant. This indicates that the nonlinear gradient is approximated better, giving better convergence results.

elements	h	$\left\|u^{-l}(\cdot, M\tau) - U_h^M\right\|_{L^2(\Gamma_h^0)}$	eoc
122	0.728614	0.22316	-
488	0.426049	0.05596	2.57788246
1952	0.265459	0.01391	2.941935144
7808	0.142662	0.00363	2.165347437
31232	0.0725927	0.00091	2.053027213
124928	0.0364637	0.00023	2.002417315
elements	h	$\left\|\nabla_{\Gamma_h^0}(u^{-l}(\cdot, M\tau) - U_h^M)\right\|_{L^2(\Gamma_h^0)}$	eoc
122	0.728614	1.31143	-
488	0.426049	0.679296	1.225920683
1952	0.265459	0.343802	1.439441454
7808	0.142662	0.172436	1.111206971
31232	0.0725927	0.0864131	1.022606166
124928	0.0364637	0.0432325	1.005808806

Table 4.3: Experimental order of convergence for $\mathbf{u}$ using Example 4.2.2

elements	h	$\left\|w^{-l}(\cdot, M\tau) - W_h^M\right\|_{L^2(\Gamma_h^0)}$	eoc
122	0.728614	3.97865	-
488	0.426049	2.30796	1.014888119
1952	0.265459	0.737787	2.410650374
7808	0.142662	0.208029	2.0386698
31232	0.0725927	0.0541002	1.993505091
124928	0.0364637	0.0137554	1.988834703
elements	h	$\left\|\nabla_{\Gamma_h^0}(w^{-l}(\cdot, M\tau) - W_h^M)\right\|_{L^2(\Gamma_h^0)}$	eoc
122	0.728614	60.2274	-
488	0.426049	54.5696	0.1838470632
1952	0.265459	34.082	0.9949557705
7808	0.142662	18.839	0.9546814543
31232	0.0725927	9.6492	0.9902908271
124928	0.0364637	4.86212	0.9954299902

Table 4.4: Experimental order of convergence for $\mathbf{w}$ using Example 4.2.2

4.3 Investigating the Parameter Space

Next we illustrate the behaviour of Problem 3.10.1 with various computational experiments. Then we discuss how this relates to the literature with examples. In particular, we give evidence of a relationship between the linker strength and the pressure. That when linker strength is increased, bleb size decreases and when pressure is increased, the membrane expands more with a greater number of linkers becoming detached. Finally, we round off the chapter by comparing our model with the one in [7] which exists for 2 dimensions. To compare we then change our model to show it is a consistent extension from this 2 dimensional model and give a numerical comparison to support this claim.

We now run examples considering two main cases. For both of these examples, we check how parameter values affect the bleb and how the surface changes over time. The main focus of parameter exploration is to look at the relationship between $\lambda_{l,\psi}$, $\lambda_{p,\psi}$ and x_0. Understanding these will indicate how the linkers are put under strain and what affects bleb growth and size. Note all parameters are only valid for the particular geometry. For example, making concave regions larger or smaller is likely to have an effect on the deformation even if only the volume is changed. These two shapes are the sphere with a protrusion which is a 3D extension of the example used in [7, 38]. Second is a biconcave shape that is a simple example of a cell with concave regions. The cortex within these surfaces we define as $\mathbf{u}_c(\mathbf{x}) := \mathbf{x} - l_0\boldsymbol{\nu}_{\Gamma^0}$.

To present all simulation results we use ParaView [3]. Note the domain Γ_h^0 itself does not evolve. Visually it is more intuitive to show the resulting image defined by $\mathbf{U}_h^m$. We hence define for each $m = 0, 1, ..., M-1$, a surface Y_h^m created by taking Γ_h^0 and replacing each vertex with the corresponding degree of freedom value $\mathbf{U}_h^m$. Often in figures of the solution we will show a vertical slice of the surface.

4.3.1 Simulations

Now we discuss and present the simulations. Comparisons use slices of the two images to see the differences more easily. The meshes were created by mapping a spherical grid under a chosen function, for more details see Section 4.5.1. All examples are carried out using a laptop with an i5-4300 processor which took several hours to complete any example in the following section.

Blood Cell

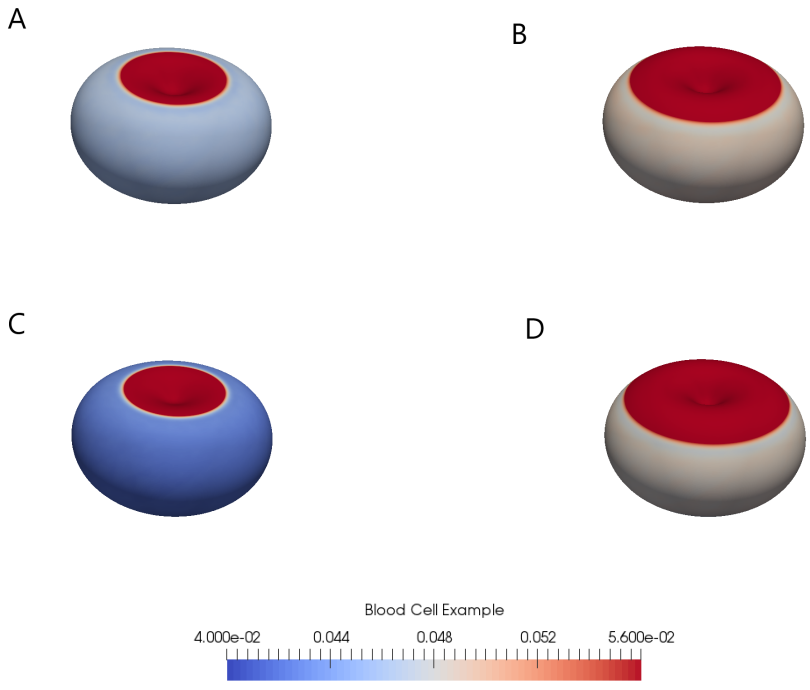


Figure 4.3: Each surface shows the final image for different sets of parameters stated in Table 4.5 and discussed in Section 4.3.1. All show the solution at time $T = 2$. The colour scheme corresponds to the strength of the connection between the membrane and the cortex. Value of $|\mathbf{u} - \mathbf{u}_c| \leq l_0 = 0.04$ is highlighted blue and $|\mathbf{u} - \mathbf{u}_c| \geq u_B = 0.056$ red. Values between this are shaded on a colour gradient.

	x_0	$\lambda_{l,\psi}$	$\lambda_{p,\psi}$	$\lambda_{b,\psi}$	l_0	u_B	τ	T	elements
Example A	0.95	18	22.5	0.005	0.04	0.056	0.0025	2	196608
Example B	0.95	12	22.5	0.005	0.04	0.056	0.0025	2	196608
Example C	0.85	18	22.5	0.005	0.04	0.056	0.0025	2	196608
Example D	0.95	18	30	0.005	0.04	0.056	0.0025	2	196608

Table 4.5: The parameters for the blood cell examples used in Section 4.3.1. Example A provides a base case which the remaining examples are compared to.

We analyse the parameters by taking one test case and measuring deviations from it by changing certain parameters and judging their effects. We create a discocyte shape by deforming the sphere of radius 4. The z component is then changed to produce the required shape. The full list of examples with their parameters are given in Table 4.5. In example A (see Figure 4.4) we see the concave regions expand outwards and lead to blebs. The detachment region eventually stabilizes preventing further deformation.

By decreasing the linker length to $\lambda_{l,\psi} = 12$, we observe more unconstrained bleb growth with more detachment (see Figure 4.5). Lower linker strength leads to a wider area of the cortex becoming separated. Once the surface is detached, the linker force plays no role, so it does not change how much the surface behaves past this however. This example is best seen compared to Figure 4.6 which increased pressure strength to $\lambda_{p,\psi} = 30$. Now the bleb's growth seems to be faster. This indicates that $\lambda_{l,\psi}$ and $\lambda_{p,\psi}$ have opposing dynamics, which one might expect.

The final case we look at is decreasing x_0 in Figure 4.7. This translates to an increase in the size of our tension force. Now we see the indentation being repelled outwards. This is supports the idea from [7] that tension plays a key role in the location of blebs. Nowhere in our model do we artificially induce a bleb so this behaviour arising from only the force balance is good for our model approach.

Spherical Protrusion

	x_0	$\lambda_{l,\psi}$	$\lambda_{p,\psi}$	$\lambda_{b,\psi}$	l_0	u_B	τ	T	elements
Example E	0.95	14	20	0.0025	0.04	0.056	0.005	3	196608
Example F	0.95	10	20	0.0025	0.04	0.056	0.005	3	196608
Example G	1.05	14	20	0.0025	0.04	0.056	0.005	3	196608
Example H	0.95	14	30	0.0025	0.04	0.056	0.005	3	196608

Table 4.6: The parameters for the spherical protrusion examples used in Section 4.3.1. Example E provides a base case which the remaining examples are compared to.

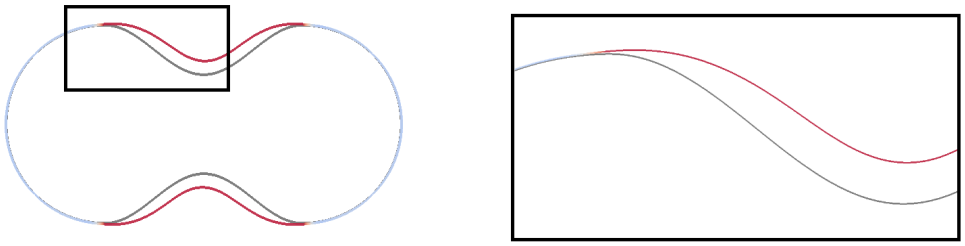


Figure 4.4: We initialise the simulation with the shape of a red blood cell as discussed in Section 4.3.1. The image shows a slice of the resulting $\mathbf{u}$ at final time overlaid on Γ_h^0 (grey curve). We use dark blue regions to highlight still attached regions, where $|\mathbf{u} - \mathbf{u}_c| < l_0$, while regions with broken linkers are coloured red ones, that is $|\mathbf{u} - \mathbf{u}_c| > u_B$. We see only a difference in the concave part, the solution has expanded outwards and detached hence is now red. A magnified image of the black box is presented on the right.

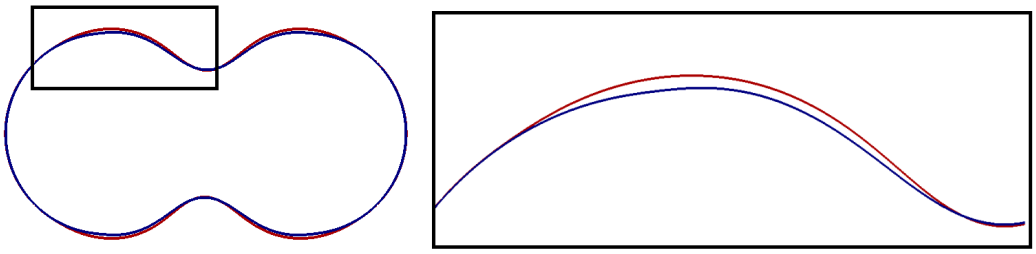


Figure 4.5: Comparison of two simulations from Section 4.3.1 that differ in the linker strength, using $\lambda_{l,\psi} = 14$ (blue) and $\lambda_{l,\psi} = 10$ (red). We do not see much difference in peak size of $|\mathbf{u} - \mathbf{u}_c|$ as once the membrane is detached the linker term with $\lambda_{l,\psi}$ vanishes. The smaller value of $\lambda_{l,\psi}$ does however produce a wider bleb site as the surface is less able to resist detached regions pulling away neighbouring areas. A magnified image of the black box is presented on the right.

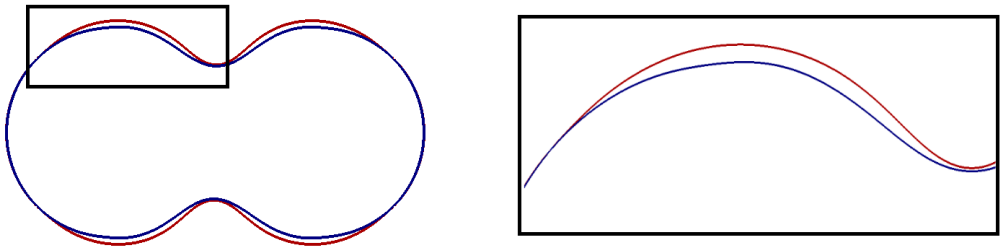


Figure 4.6: Comparison of two simulations from Section 4.3.1 that differ in the pressure, using $\lambda_{p,\psi} = 22.5$ (blue) and $\lambda_{p,\psi} = 30$ (red). Once the surface is detached from the linkers ($|\mathbf{u}-\mathbf{u}_c| > u_B$), it now moves further due to the increased $\lambda_{p,\psi}$. The bleb now breaks a wider area of linkers as we see deformation much further away from the initial bleb site in the concave region. A magnified image of the black box is presented on the right.

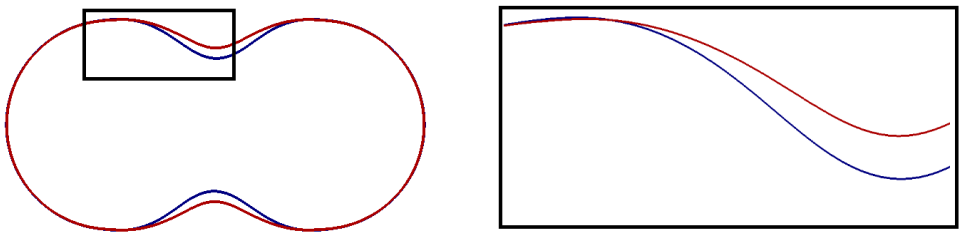


Figure 4.7: Simulation run that increase the importance of tension as discussed in Section 4.3.1. For the blue curve uses $x_0 = 0.95$ and for the red curve $x_0 = 0.85$. In the concave neck region the lower x_0 pulls the membrane away from the cortex while the rest of the membrane retracts so $x_0 = 0.85$ is slightly inside the blue curve foe 0.95 in area of positive curvature. A magnified image of the black box is presented on the right.

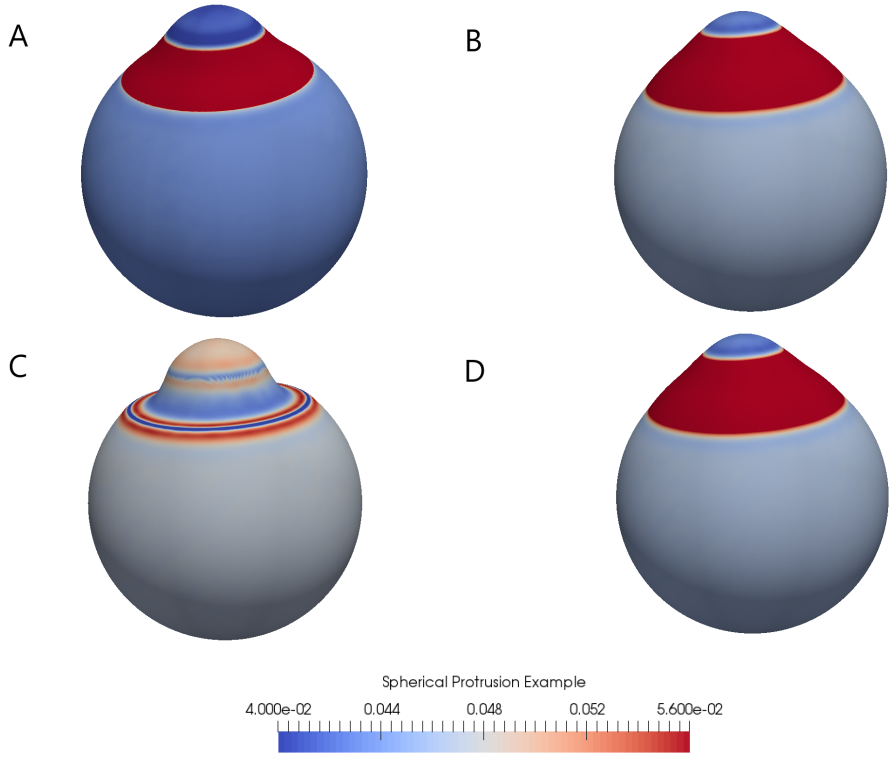


Figure 4.8: Each surface shows the final image for different parameter sets as stated in Table 4.6 and discussed in Section 4.3.1. All show the solution at time $T = 3$. The colour scheme indicates the strength of the connection between the membrane and the cortex. Value of $|\mathbf{u} - \mathbf{u}_c| \leq l_0 = 0.04$ is highlighted blue and $|\mathbf{u} - \mathbf{u}_c| \geq u_B = 0.056$ red. Values between this are shaded on a colour gradient.

For the second example, we take a sphere with a protrusion as this is a symmetric extension of an example often used in 2D [7]. For this new shape we project to a sphere of radius 4 and for the points such that $z \geq 3.1225$ are then additionally deformed to create the protrusion. As our model posses no way to break the symmetry, any blebs should also be symmetric. In practise numerical noise may create some asymmetry but the deformation appears stable. Again we work with a test case and investigate changes from this base case. Now in example E (from Table 4.6) we see a ring-like protrusion which is radially symmetric. The breaking of linkers does stop or at least slows down to very slow speeds. In this example, we do see a bleb form in the predicted location but no balloon-like structure is formed.

In Figure 4.10 we set $\lambda_{p,\psi} = 30$ and compare to example E. Here we can see the resulting bleb expands. In regions not containing a bleb, such as the sphere's

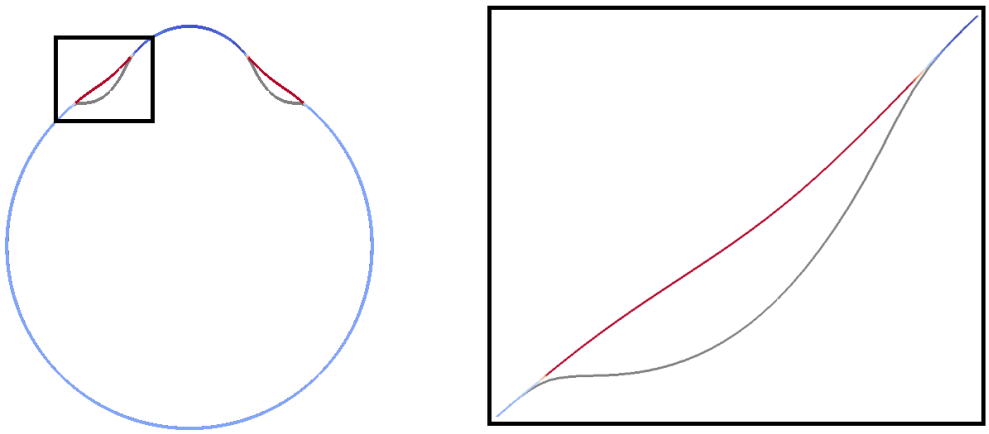


Figure 4.9: We initialise the simulation with the shape of sphere with a protrusion. The image shows a slice of the resulting $\mathbf{u}$ at final time overlaid on Γ_h^0 (grey curve) from Section 4.3.1. Dark blue regions highlight attached regions, where $|\mathbf{u}-\mathbf{u}_c| < l_0$, while red ones correspond to regions with broken linkers, that is $|\mathbf{u}-\mathbf{u}_c| > u_B$. We see that the concave part of the cell expands outwards. A magnified image of the black box is presented on the right.

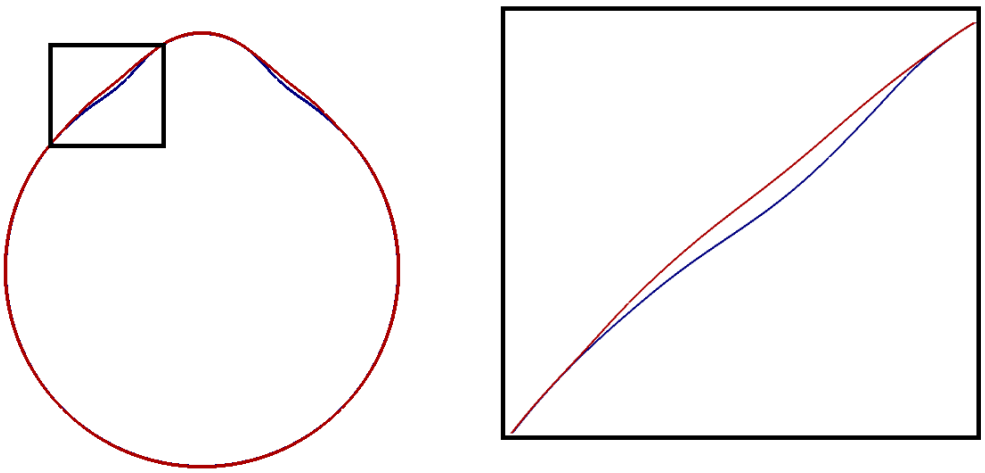


Figure 4.10: Comparison of membranes using $\lambda_{p,\psi} = 20$ (blue) and $\lambda_{p,\psi} = 30$ (red) from Section 4.3.1. The surface with higher $\lambda_{p,\psi}$ moves further away from the cortex and this extra movement helps detach more of the linkers. A magnified image of the black box is presented on the right.

underside is under much more strain as indicated by a paler blue colour (see Figure 4.8). Again we contrast this with the comparison of variation in the linker strength. In Figure 4.11 we use an example with $\lambda_{l,\psi} = 10$, we see that a stronger linker force has the opposite effect to $\lambda_{p,\psi}$. Now the surface with smaller $\lambda_{l,\psi}$ has larger blebs and linkers on the underside are further from the cortex. This adds credence that the two terms are in contention and either the pressure dominates leading to the membrane ripping or the linkers are stronger which causes more difficulty in propagating a bleb.

Now we look to check the role of tension by taking $x_0 = 1.05$ so the surface now looks to increase its area instead of decreasing. This reverses the direction of the tension force. In Figure 4.12 we see only a small bleb. This suggests tension can be a key is propagating bleb sites and how they evolve. Here we have no restriction of area so no force pulls convex region.

4.3.2 Simulation Conclusions

We do not need to nucleate our blebs in any way such as weakening linkers or removing them. In [7] a known issue was blebs propagating without limit. That does not seem to be an issue here. While we did not see blebs reach a stationary point, the speed did became very slow in the example without high $\lambda_{p,\psi}$. We argue

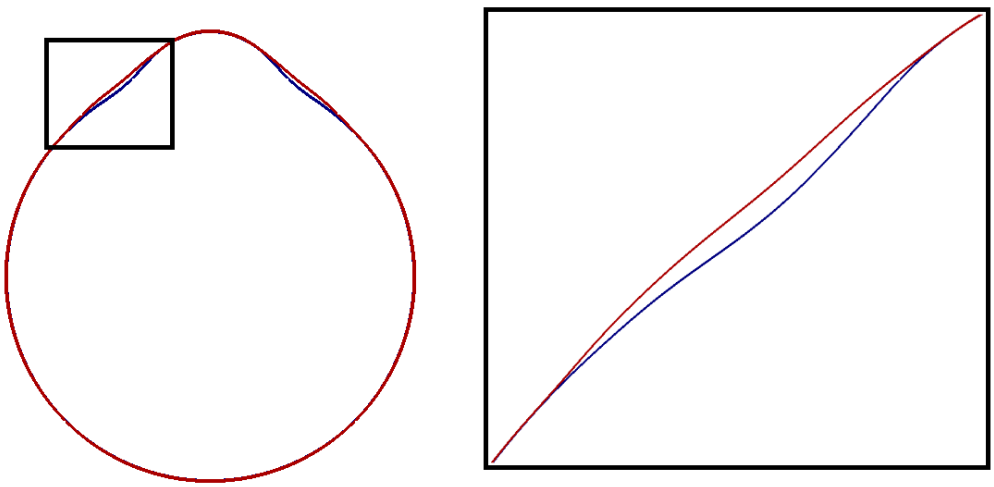


Figure 4.11: Comparison of membranes under different $\lambda_{l,\psi}$ overlaid and discussed in Section 4.3.1. The difference looks similar to Figure 4.10 but now the surface is moved in the opposite direction. This suggests that the parameters $\lambda_{l,\psi}$ and $\lambda_{p,\psi}$ work towards opposing goals as expected.

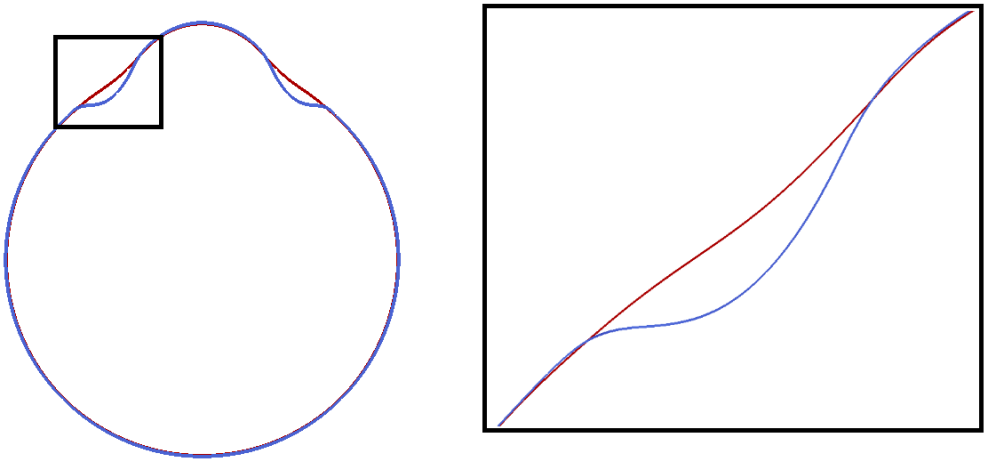


Figure 4.12: Comparison of two simulations that differ on the tension as discussed in Section 4.3.1, using $x_0 = 1.05$ (blue) and $x_0 = 0.95$ (red). Here the blue surface has not moved at all. This suggests that when the surface is not under tension $x_0 > 1$ that blebs do no form in our model. This means the tensions is working as the driving force of blebs in our model. A magnified image of the black box is presented on the right.

that at longer time scales the cortex attempts to reattach to the membrane during the next stage of the bleb cell cycle (see Figure 1.1). This is assuming our parameters are somewhat accurate. We argue this blebbing only exists within a narrow range of parameters when the bleb growth is not unstrained and linkers are weak enough to be broken.

The examples covered indicate that we are able to capture some blebbing in these simple 3D examples. The main strength of this approach lies in the ability to trial different shapes and keep the same model. This is in stark contrast to existing models that could only handle 2D shapes or radial symmetry in 3D. As we will show at the end of the chapter, our scheme can handle triangulations from realistic surfaces. The implementation could be given to those of greater biological knowledge to run their own examples. We can see certain weaknesses in the model. We see no balloon-like protrusions that are seen in the expansion stage of blebbing (see Figure 1.1). The culprit for this could be any of the linker function $\mathbf{k}$, the pressure or some aspect missing by the model. Bonds could be changed to break at different rates. This would affect the area of linkers broken would not lead to balloon-like shapes as once a linker is broken in plays no further role. This is somewhat surprising as we modelled linkers as linear springs (2.12) with smoothened discrete jumps (2.18). The pressure is likely a key factor as when $|\mathbf{u} - \mathbf{u}_c| > u_B$ the linker term vanishes so the pressure should dominate and inflate the region. Using a fixed normal $\boldsymbol{\nu}_{\Gamma^0}$ however makes it unlikely areas of positive curvature form from regions of negative curvature. This is because it would require the bending force to dominate which the literature indicates is small. Using a evolving normal this may however happen.

4.4 Comparison to an existing model

We now wish to compare Problem 2.5.1 with the 2D model found in [7]. This shows how the two models match up despite using different discretisations. In our approach, we have used FEM with a formulation over the surface Γ^0 in 3D. This differs from the 2D finite difference method (1.1) that parametrises the initial surface. In this section, we reduce the dimension of our model to create a "strip" model and ran comparisons accounting for the different parametrization. As the model in [7] is dimensional we include the required coefficient to match for this section. This means we can compare to their grid sizes and outputs directly. The underlining linker force is the same in both models so we set $k_l = 0$ for all comparisons to better test the remain terms.

4.4.1 Model comparison and adaptation

To our knowledge Dune does not contain functionality for curves to be used as domains in 2D space. Instead, we extend the problem in the z direction to create a version such that the $x-y$ plane solves the original problem and is only constant in the third direction. This allows us to reuse the existing software implementation. For clarity we state the underlining equation found in [7] which underpins finite deference scheme (with $k_l = 0$, linker term dropped for simplicity).

Problem 4.4.1. *Underlining model of [7]. For given constants $\omega, p, k_b, k_\psi, x_0$, find $\mathbf{u} : \Gamma^0 \times [0,T) \to \mathbb{R}^2$ such that*

$$\omega \partial_t \mathbf{u} + k_b \Delta^2_{\Gamma^0} \mathbf{u} + k_\psi \nabla_{\Gamma^0} \cdot \left(\nabla_{\Gamma^0} \mathbf{u} - x_0 \frac{\nabla_{\Gamma^0} \mathbf{u}}{|\nabla_{\Gamma^0} \mathbf{u}|} \right) + \frac{p}{|\Omega|} \boldsymbol{\nu}_\Gamma = 0 \tag{4.11}$$

such that $\hat{\mathbf{u}}(\cdot, 0) = \mathbf{id}_{\Gamma^0}$ and the image of $\mathbf{u}$ be the curve $\Gamma(t) := \mathbf{u}(\Gamma^0, t)$ with outward normal $\boldsymbol{\nu}_\Gamma$ and enclosed area Ω.

Note that scheme in [7] uses a parametrisation to a circle of discrete length N which is divided into N unit intervals. When comparing between schemes this can be accounted of by adjusting the parameters.

Remark 4.4.1. *For a parametrisation $\mathbf{p} : S^1 \to \Gamma^0$ we write $\mathbf{u}(\mathbf{x}) = \mathbf{u}(\mathbf{p}(\gamma))$ such that $|\mathbf{p}'(\gamma)| = l$ for some constant l. This allows us write*

$$\omega \partial_t \mathbf{u} + \frac{k_b}{l^4} \Delta^4_{S^1} \mathbf{u} + \frac{k_\psi}{l^2} \nabla_{S^1} \left(\left(\mathbf{I} - \frac{x_0}{l |\nabla_{S^1} \mathbf{u}|} \right) \nabla_{S^1} \mathbf{u} \right) + \frac{p}{\Omega(\mathbf{u})} \boldsymbol{\nu}_\Gamma = 0$$

Our approach is to create a constant extension in the z-axis for the domain and restricting the nonlinear gradient term. As our model is posed over Γ^0_h and not S^2 we also pick up parametrisation terms. In [7] they parametrise a length N discrete curve by N unit intervals so we assume our parametrisation is also uniform. This means the rate of change in the parametrisation is constant. When such term appear we choose to absorb them into that terms coefficient. As no abstract theory underpins [7] they did need to include any ϵ_i values to avoid discontinuous jumps as we did in Section 2.4. We now take $\epsilon_i = 0$ for $i = 1, 2, 3$ to make comparisons easier . Also, the pressure term is now assumed to use the normal and volume from the moving surface to better match the two models.

The domain is now $\tilde{\Gamma}^0 := \Gamma^0 \times [-0.25, 0.25]$ such that Γ^0 is the initial 2D curve of the membrane (see Figure 4.13). The functions $\mathbf{u}, \mathbf{w}$ must also be extended to three dimensional vector valued functions, this is done by defining a z position u_3 in

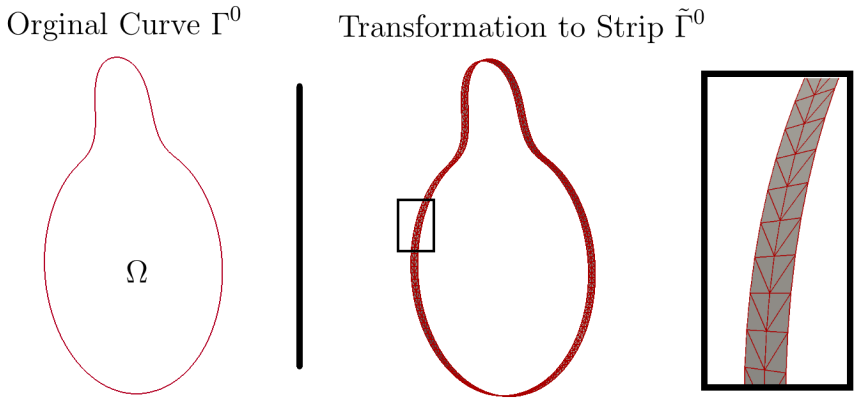


Figure 4.13: The 2D curve Γ with area Ω is transformed into a strip by adding point in the $\pm z$ direction. This new strip $\tilde{\Gamma}^0$ is the domain used for our estimation of the finite difference scheme used in [7]. A zoomed in picture of the black box is displayed on the right.

the case of $\mathbf{u}$ and likewise for $\mathbf{w}$. Let $\tilde{\mathbf{u}} = (u_1, u_2, u_3) \in \hat{\Gamma}^0$ be the solution on the strip that we use to recover the 2D model, likewise for $\mathbf{w}$. Then we let $\mathbf{u} = (u_1, u_2) : \mathbb{R}^3 \to \mathbb{R}^2$ be the projection of $\tilde{\mathbf{u}} : \mathbb{R}^3 \to \mathbb{R}^3$ to the $x - y$ plane. Using similar notation for test function we define three dimensional functions $\boldsymbol{\varphi} = (\varphi_1, \varphi_2, \varphi_3), \boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3)$ with restrictions to the $x - y$ plane as $\tilde{\boldsymbol{\varphi}} = (\varphi_1, \varphi_2), \tilde{\boldsymbol{\eta}} = (\eta_1, \eta_2)$. The new weak formulation is then

Problem 4.4.2. *(Strip formulation) Define the 2D curve Γ^0 in the $x - y$ plane with constant extension to a strip $\tilde{\Gamma}^0 := \Gamma^0 \times [-0.25, 0.25]$. Denote $\tilde{\mathbf{u}} := (\mathbf{u}, u_3)$ and $\tilde{\mathbf{w}} := (\mathbf{w}, w_3)$. Then, for given constants $\omega, p, k_b, k_\psi, x_0$ and given function $\mathbf{u}_c : \tilde{\Gamma}^0 \to \mathbb{R}^3$, find $(\mathbf{u}, \mathbf{w}) : \tilde{\Gamma}^0 \times [0, T) \to \mathbb{R}^6$ such that for all $\boldsymbol{\varphi}, \boldsymbol{\eta} \in [H^1]^3$ and all t*

$$\int_{\tilde{\Gamma}^0} \omega \partial_t \tilde{\mathbf{u}} \tilde{\boldsymbol{\varphi}} + k_b \nabla_{\tilde{\Gamma}^0} \mathbf{w} \nabla_{\tilde{\Gamma}^0} \boldsymbol{\varphi} + k_\psi \left(\nabla_{\tilde{\Gamma}^0} \mathbf{u} - x_0 \frac{\nabla_{\tilde{\Gamma}^0} \mathbf{u}}{|\nabla_{\tilde{\Gamma}^0} \mathbf{u}|} \right) \nabla_{\tilde{\Gamma}^0} \boldsymbol{\varphi} + \frac{p}{|\Omega|} \boldsymbol{\nu}_{\tilde{\Gamma}} \tilde{\boldsymbol{\varphi}} d\tilde{\sigma} = 0 \tag{4.12}$$

$$\int_{\tilde{\Gamma}^0} \tilde{\mathbf{w}} \tilde{\boldsymbol{\eta}} - \nabla_{\tilde{\Gamma}^0_h} \mathbf{u} \nabla_{\tilde{\Gamma}^0_h} \boldsymbol{\eta} d\tilde{\sigma} = 0 \tag{4.13}$$

such that $\tilde{\mathbf{u}}(\mathbf{x}, 0) = \mathbf{x}$, $\mathbf{x} \in \Gamma^0$ and boundary conditions

$$\nabla_{\Gamma^0} u_3((x, y, \pm 0.25), t) \cdot \mathbf{e}_z = \nabla_{\Gamma^0} w_3((\cdot, \pm 0.25), t) \cdot \mathbf{e}_z = 0 \tag{4.14}$$

Lemma 4.4.1. *The solution to Problem 4.4.2 is invariant in its z coordinate for any $z \in [-0.25, 0.25]$, the solution $\mathbf{u} = (u_1, u_2)$ in $x - y$ plane is equivalent to the solution of Problem 4.4.1 up to a parametrization.*

Proof. For most terms in the model the third components has been removed. Other terms such as the $\boldsymbol{\nu}_{\tilde{\Gamma}}$ point in the $x - y$ plane which is orthogonal by construction at $t = 0$ and also $w_3 = 0$ by similar reasoning. These terms will remain 0 in z unless changed by an additional term. As we see our claim holds for the base case we need only show that if it holds for current time all future times will act orthogonal to z. Assuming this, all that remains is the first term for motion in x_3 direction, i.e for functions we use the notation $\tilde{\boldsymbol{\varphi}} = (\varphi_1, \varphi_2, \varphi_3)$. Choosing $\tilde{\boldsymbol{\varphi}} = (0, 0, \varphi_3)$ we see that

$$\int_{\tilde{\Gamma}^0} \omega \partial_t u_3 \varphi_3 d\tilde{\sigma} = 0.$$

This means that movement in the z direction is zero for all time. We can now restrict our test space to functions $(\varphi_1, \varphi_2, 0)$ without loss of generality, denoting $\boldsymbol{\nu}_\Gamma = (\nu_{\Gamma,1}, \nu_{\Gamma,2})$ we have in terms of two dimensional functions

$$\int_{\tilde{\Gamma}^0} \omega \partial_t \mathbf{u}\boldsymbol{\varphi} + k_b \nabla_{\tilde{\Gamma}^0} \mathbf{w} \nabla_{\tilde{\Gamma}^0} \boldsymbol{\varphi} + k_\psi \left(\nabla_{\tilde{\Gamma}^0} \mathbf{u} - x_0 \frac{\nabla_{\tilde{\Gamma}^0} \mathbf{u}}{|\nabla_{\tilde{\Gamma}^0} \mathbf{u}|} \right) \nabla_{\tilde{\Gamma}^0} \boldsymbol{\varphi} + \frac{p}{|\Omega|} \boldsymbol{\nu}_\Gamma \boldsymbol{\varphi} d\tilde{\sigma} = 0, \tag{4.15}$$

$$\int_{\tilde{\Gamma}^0} \mathbf{w}\boldsymbol{\eta} - \nabla_{\tilde{\Gamma}^0_h} \mathbf{u} \nabla_{\tilde{\Gamma}^0_h} \boldsymbol{\eta} d\tilde{\sigma} = 0. \tag{4.16}$$

As no movement or change happens in the z direction for the $x - y$ coordinates we can separate out the integral into Γ^0 and the extension.

$$\int_{-0.25}^{0.25} \int_{\Gamma^0} \omega \partial_t \mathbf{u}\boldsymbol{\varphi} + k_b \nabla_{\Gamma^0} \mathbf{w} \nabla_{\Gamma^0} \boldsymbol{\varphi} + k_\psi \left(\nabla_{\Gamma^0} \mathbf{u} - x_0 \frac{\nabla_{\Gamma^0} \mathbf{u}}{|\nabla_{\Gamma^0} \mathbf{u}|} \right) \nabla_{\Gamma^0} \boldsymbol{\varphi} + \frac{p}{|\Omega|} \boldsymbol{\nu}_\Gamma \boldsymbol{\varphi} d\sigma dz = 0$$

$$\int_{-0.25}^{0.25} \int_{\Gamma^0} \mathbf{w}\boldsymbol{\eta} - \nabla_{\Gamma^0} \mathbf{u} \nabla_{\Gamma^0} \boldsymbol{\eta} d\sigma dz = 0$$

We can in fact drop the integral over z and conclude that we recover the weak form of Problem 4.4.1 with an operator splitting method.

□

To implement Problem 4.4.2 we must alter the system matrix as described

in (3.68). This is done by removing the z component of the matrix in all but the mass matrices of the forward and backwards steps of the time stepping. These two terms form a identity block that stores the initial z values of the points. The mass matrix now reads

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}^L & 0 & 0 \\ 0 & \mathbf{M}^L & 0 \\ 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{3N\times 3N} \text{ where } \mathbf{M}^L = \left\{ \int_{\tilde{\Gamma}_h^0} \phi_i(\mathbf{x})\phi_j(\mathbf{x}) \right\}_{i,j=1}^{N}, \quad (4.17)$$

with similar structure for the stiffness matrix. In practise the z components can be removed to save on computation size of the matrix.

4.4.2 Numerical comparison

We aim to show that we can recreate with our FEM scheme (Problem 4.4.2) the same model seen in [7] that uses finite difference and was completed in MATLAB. Their code was made available to us so results can then be compared to show that we recover a unique solution.

An expected difference is in the parametrization. In the MATLAB code the curve Γ^0 is divided into N segment of unequal length. Each section is parametrised by a unit length interval which means the parametrisation's speed is not constant. As the formula for the parametrisation is not known as we have only grid points we instead use an averaging to estimate the projection's value. The finite element scheme uses a first order time stepping method and finite difference scheme used a three-stage, third-order, Runge-Kutta method. To minimise this difference we used a quarter the size time step τ during the FEM compared to the finite difference scheme.

The Γ^0 grid file comes from a transformation of the mat file used in the MATLAB code to DUNE compatible format. The triangulation is made by adding points above and below the points and mesh appropriately. For example the point (x, y) generates three points in total. First we extend it to the 3D world by writing $(x, y, 0)$, then we add a point above $(x, y, 0.25)$ and below $(x, y, -0.25)$. These then form the nodes for a triangulation.

The parameters used in both formulations are $\omega = 2.1677 \times 10^{-8}$, $x_0 = 0.2922$, $p = 10^{-5}$, $k_b = 0.07 \times 10^{-6}$and $k_\psi = 8.5 \times 10^{-6}$. In the case of Problem 4.4.2 a parametrisation was also accounted for. The time step in the MATLAB code is $\tau = 0.002$ and in the DUNE version $\tau = 0.0005$ to minimise the difference from the different order time discretisation.

As expected the cell simply expands outwards with the concave part being

removed (Figure 4.14). We compare the discrete volumes of each example below. The volume converges as the solution $\mathbf{u}_h$ converges as they both depend on the convergence of the distance between a discrete surface Γ_h and Γ [15]. This also allows for easier comparison of results that use different software and hence differ with respect to the data structures for the solution $\mathbf{u}$. In the finite element case, we measure the area enclosed by our strip and the $z = 0$ plane.

While the relative error does increase over the time interval we are simulating, it remains less than 10^{-4} so for our purposes it does, in fact, recover the same solution to the problem. A visual comparison showing how close the solutions are can be seen in Figure 4.15.

Timestep	Matlab	FEM	Difference	Relative error
0.1	175.9015	175.8951	0.0064	0.00003638532284
0.2	186.1317	186.1221	0.0096	0.00005157904408
0.3	195.4352	195.4236	0.0116	0.00005935823514
0.4	203.9965	203.9835	0.013	0.00006373064488
0.5	211.9348	211.9197	0.0151	0.000071253404
0.6	219.3335	219.3156	0.0179	0.00008161754111
0.7	226.2476	226.227	0.0206	0.00009105898058
0.8	232.7126	232.6899	0.0227	0.00009755472842

Table 4.7: A comparison between the finite difference scheme completed in MATLAB and the finite element scheme implemented in Dune. The small relative error suggest the FE scheme is an accurate recreations.

4.5 Using Real Data

The core interest of this work to a nonmathematician is the ability to use triangulations based on cell images and the option to change parts of the model. Having demonstrated what is possible with the scheme we now discuss these aspects and outline how it could be used by new users. The goal we have in mind is someone of a biological background who wished to carry out simulations and produce examples of their own. For example, artificial geometries of their choice or of real cells. In this section we taking existing triangulation that is based on cell image data and show that our scheme can handle such meshes. We then showcase an implementation of our scheme using the new Python frontend to Dune called Dune-FemPy.

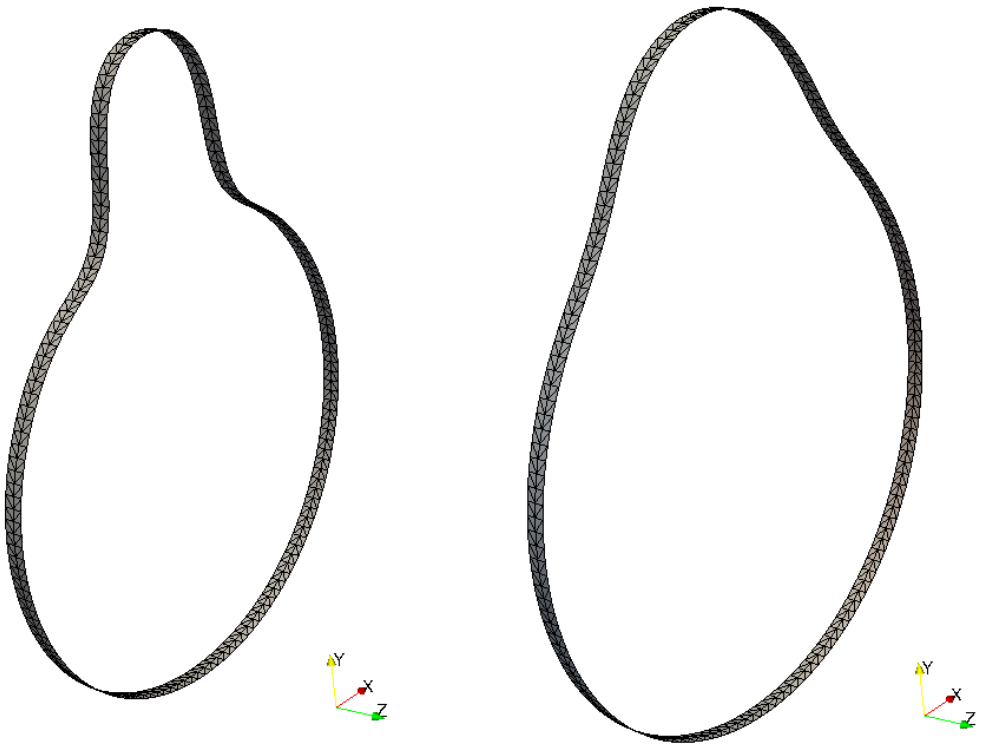


Figure 4.14: On the left is Γ_h^0 and on the right is the image of $\mathbf{u}$ at $t = 0.8$. The parameters used are $\omega = 2.1677 \times 10^{-8}$, $x_0 = 0.2922$, $p = 10^{-5}$, $k_b = 0.07 \times 10^{-6}$and $k_\psi = 8.5 \times 10^{-6}$. To create the initial strip we started with a discrete curve and added a set of vertices on either side in the z direction to give rise to a thin mesh.

Figure 4.15: Here is the surface given by the FE scheme at $t = 0.8$ in grey. Imposed on the image is a black curve representing the FD result at that time. As suggested by the low relative error in Table 4.7, the value in the $x - y$ plane are indistinguishable.

To create the grid that is readable by the code we must make a dgf file which is the standard grid file for Dune. This contains three main pieces of information, a list of vertices, the labels of each vertex in an element and any projection rule. The existing triangulation was given in a format only readable by MATLAB so requires conversion. The image's triangulation was produced by using [5]. To produce Figure 4.16 we read information in via a MATLAB script which reads the MATLAB mesh file and outputs a dgf file in the correct format.

In Figure 4.16 we run the scheme on some real data, the format of deforming the grid and using a colour scheme remains. For parameters we chose $\lambda_{b,\psi} = 0.01$, $\lambda_{l,\psi} = 100$, $\lambda_{p,\psi} = 20$, $x_0 = 0.95$, $l_0 = 0.04$, $u_B = 0.056$, $k_L = 500$ and $u_{com} = 0.03$. Here we have kept $\lambda_{b,\psi}$ low as with previous examples, $\lambda_{l,\psi}$ and $\lambda_{p,\psi}$ which are sizes similar to simulation carried out in [38]. Parameters used in [38] applied here may not be inaccurate for this case however. Because of this we do not claim any quantitative results from these simulations. In Figure 4.16 no blebs are induced manually, in no regions are the linker connections weakened. All the detachments are formed by the tension and pressure forces. In practice other reasons may drive blebbing such as some internal cell process.

The surface breaks from the cortex in two main ways. On the positive side, we do see breakage in a concave region as expected. Focusing on the main bulk of the cell away from the flattened regions this seems to hold true. The other regions of red that we see are away from this bulk seem to be at the cells sides where sharp protrusions form. Here the model seems to break down when posed over a very small protrusion. The difficulty in preventing this is strongly linked to how one constructs the cortex. A triangulation of the surface contains no information on the cells internal structure however. Currently over time these contract into sharp points. This should be prevented by the bending but in blebbing models this force should be small.

4.5.1 Dune FemPy

We have implemented a version of our scheme in a Python front for Dune-Fem called Dune-FemPy. This reduces our scheme to a short Python script that gives a more user-friendly experience that reduces the burden of knowledge for changing the problem. This comes at a cost of speed in certain cases. If the Python side of the program is called within the time loop this can slow down the runtime. To finish this chapter we explain our implementation and highlight some of the features.

For this example, we limit our scope to reading in grids, the deformations, parameters and formulating the equation. These are the most useful to a new user

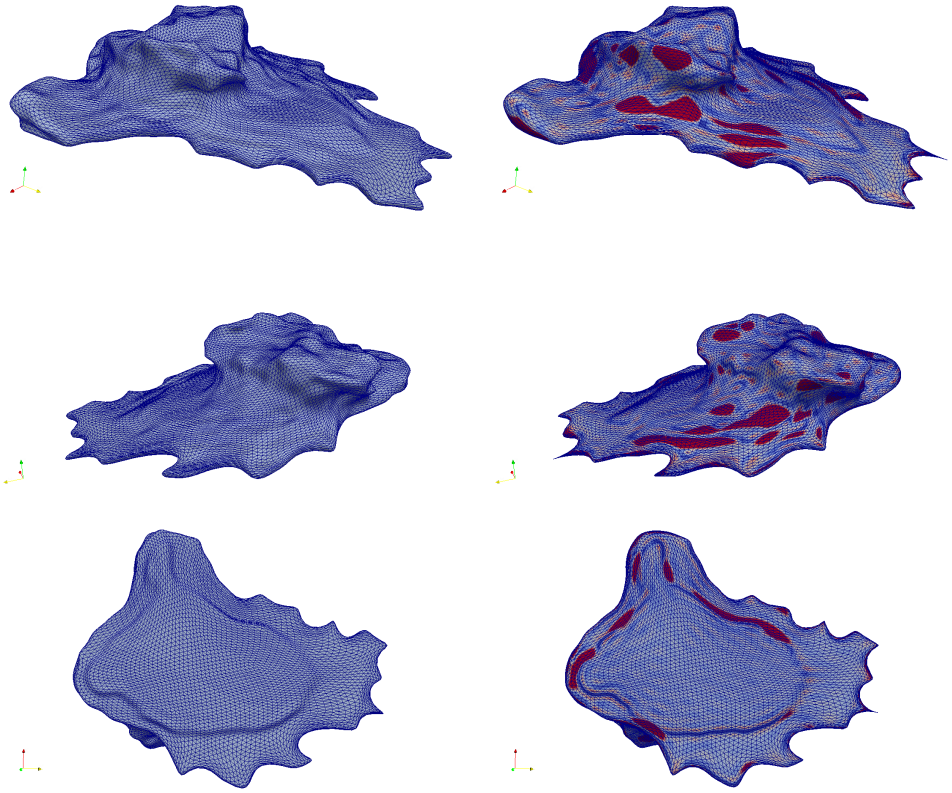


Figure 4.16: Application of proposed scheme for different cell images as discussed in Section 4.5, blebs seem to form in some concave region but also seem to form from ripples/noise in the surface. Some small protrusion at the edge contract to sharp points.

for understanding the code.

```
gridSelect = "cell.dgf"
# Select how you wish to deform your grid (dgf file)
case = 0
Graphics = 1
```

Here we select the grid type "cell.dgf" which is our created grid file from on a cell's image data. Alternatively one can read in the grid of say a sphere. To alter the surface change the "case" value which selects how any grid can then be deformed. Finally, the graphics setting defines what information it contains in the output file. Current options include $(\mathbf{u}, \mathbf{w})$ over Γ_h^0 (Graphics= 0) or $\mathbf{u}$ on a surface morphed under $\mathbf{u}$ (Graphics= 1), and finally graphics can be turned off all together (by setting any other value).

```
surfaceOptions = { 0 : lambda x: identity(x),
1 : lambda x: Ball(x),
2 : lambda x: Bloodcell(x)
3 : lambda x: Pear(x)
}
```

We include functionality to alter grids once they are read in by the previously mentioned "case" variable. Here user selects a function to map point x to a new location. Here projection rules can be inserted. This allows for toy geometries to be set up such as those used earlier in this chapter. Which surfaceOption is used depends on the value set. Exact functions can be found in the "deformation.py" file available on my GitHub account (https://github.com/AdamNixon/Warwick. One such function is called "Bloodcell" which creates the discocyte shapes used in Section 4.3.1.

```
def Bloodcell(x):
y=[0, 0, 0]
r1 = sqrt(x[0]*x[0]+x[1]*x[1]+x[2]*x[2])
r0 = 4.
y[0] = r0*x[ 0 ]/r1
y[1] = r0*x[ 1 ]/r1

r2 = sqrt(x[0]*x[0]+x[1]*x[1])
r3 = sqrt(y[0]*y[0]+y[1]*y[1])
y[2] = 1.5-0.5*cos(pi*r3/2.)
if r3 > 2.:
y[2]=sqrt(4.-(r3-2.)*(r3-2.))
if x[2] < 0.:
```

```
y[2]=-y[2]
return y
```

We also define parameters, including those for the time discretisation, for example K_0 matches the k_l used throughout.

```
# Variables
deltaT = 0.002
finalT=  2
K_PSI = 1
K_B = 0.005
X_0 = 0.95
P_0 = 7.5
K_0 =  4
U_B = 0.056
L_0 = 0.04
Omega = 1
```

We can match each parameter by the table below.

LaTeX	fempy
τ	deltaT
T	finalT
k_ψ	K_PSI
k_b	K_B
x_0	X_0
p	P_0
k_l	K_0
u_B	U_B
l_0	L_0
ω	Omega

We now shift our focus to aspects of the code that require a more mathematical understanding of the model. First, we start with an example of a function.

```
def A1(u):
return as_vector([u[0], u[1], u[2]])
```

As the solution consists of $(\mathbf{u}, \mathbf{w})$ this function returns only $\mathbf{u}$. It has a matching function "A2" that returns $\mathbf{w}$. This is useful for controlling which parts are used. This is found in the "functions.py" file along with similar functions that help define the bilinear form. These can then be grouped into the terms that make up the model equation. Here we look at the implementation of (3.67)

```
f = inner ( grad(A1(u)) ,  grad(A2(v))  )
g = inner( A2(u), A2(v) )
```

We see a use of the A1/A2 functions mentioned earlier now also applied to the test function $\mathbf{v}$. Being able to write expressions in terms of inner products is really advantageous for ease of understanding the code. This also means the user can much easier edit or add their own functions. Once done, the different terms are simplify added together. Note for the known explicit terms we have used $\mathbf{u}_n$ which also appears in the definitions of terms with the *ex* index.

```
a_ex=(omega*inner(A1(u_n),A1(v))
        +tau*(Tension_ex+Pressure+Linkers_ex))*dx
a_im=(omega*inner(A1(u),A1(v))
        +tau*(Tension_im+Bending+Linkers_im)+g-f)*dx

equation = a_im == a_ex
```

This resembles the equation seen in Problem 3.10.1 with ex subscript for the fully explicit parts and im for the remaining mostly implicit parts. Once formulated the model class should be told of the equation.

```
model = create.model("elliptic", surface, equation,
          coefficients={u_n:solution_n,u_0:solution_0})
```

Here the model needs to know of the grid (surface) and scheme (equation) and any other terms such as the previous time step.

In this Chapter we have given various numerical examples of how Problem 3.10.1 acts as a model for cell blebbing and that it is an extension of the 2D model found in [7, 38]. At the start of the chapter we showed two example for the experimental order of convergence. Once this was completed we gave examples of our model simulating blebbing. The two test cases were of a blood cell like shape and a sphere with protrusion. These both suggested a connection between linker and pressure forces but it was the tension force that seems to cause particular bleb sites. These are restricted to concave section of the surface. Finally we show that when restricted to 2D, our model reproduces a solution with tiny relative error compared to the starting MATLAB scheme. This is evidence that our model is an extension of the work in [7, 38]. We finished this Chapter with a brief outline of our implementation.

Chapter 5

Conclusion

We finish by giving an overview of the contributions made in this thesis. The main aim was to develop, analyse and implement a three-dimensional cell blebbing model to address where the geometry of the cells cause breakage to occur during the initial stages of blebbing.

In the first part, we proposed a new 3D model for cell blebbing after reviewing the literature. The model is based on a variational formulation which includes bending, tension, linker, and pressure forces. In the literature these forces are used to derived various models [7, 38, 34, 35].

The elastic forces are given by the variation of elastic energy. We used a linear bending force as the coefficient in our model application is known to be small. This might not be the case in other applications, hence if developing a model further the bending energy could be replaced by the fully nonlinear Helfrich energy. This would lead to more complicated nonlinear higher order terms.

The membrane is initially attached to a second surface called the cortex, which is assumed to be stationary. We model this linkage by a constant linker density. In the original finite difference scheme, a discrete number of these springs exist but for our purposes we generalise. Beyond a certain distance, the linkers are considered broken and reduce its coefficient to zero. We assumed these springs worked approximately linearly while connected.

With Model 2.2.1 in place, we augment it by adding some smoothing parameters. This allows us to prove analytical results. We then concentrate on Problem 2.5.1, which is the focus of the thesis. This is a fourth order problem with a nonlinear gradient term. We solve the problem by introducing an operator splitting that allows us to use linear finite elements since the splitting leads to a second order system.

In Chapter 3 the focus is on a semi discrete scheme (Problem 3.3.2). We reposed the problem in terms of generalised functions taking certain assumptions on the growth and regularity of the nonlinearities. We managed to show weak well-posedness of the semi-discrete scheme in Theorem 3.8.1. First, we show local in time existence. Then we prove that the solution does not blow up in time. This gives together with suitable a priori estimates, global in time existence.

We can then prove convergence of Problem 3.3.2 to the continuous formulation in Problem 3.3.1. This follows by translating the discrete bounds (3.34)(3.38) to the lifted cases (3.42)(3.43) which gives us weak convergences in suitable spaces. For the non-linear terms, we are able to show the required strong convergence. This allows us to conclude the convergence of the fully discrete scheme to the limiting problem. A uniqueness proof finishes the well-posedness statement. In the case of further regularity, improved bounds can be shown. The chapter finishes with proof that the regularized cell blebbing model in Problem 2.5.1 does meet the conditions of Problem 3.3.1.

Results for this abstract problem can still hold if the model is changed as long as the assumptions on the structure and growth of the nonlinearities are not violated. It may be possible to generalise these conditions much further, but these would likely be restricted to the lowest order terms. This is still useful as it allows for alternative linker models which have different stress properties. For the gradient, convergence may still hold for the case of a non Lipschitz function but such a result is more difficult and the condition is sufficient for the function we had in mind.

Chapter 3 ends by presenting the fully discrete scheme for the regularised model (see Problem 2.5.1) and shows stability. The only terms taken explicitly are the pressure and parts of the linker and tension contributions. Once the nonlinearities are explicitly calculated, no nonlinear solver such as Newton's method is required as the problem then become linear. The stability proof is able to take advantage of the non-linear gradients structure. One extension would be to show error rates for this fully discrete scheme. The solver used could also be optimised by using the saddle point structure of the system of equation rather than the general solver used currently.

In Chapter 4 the focus of the thesis shifts from analysis to numerical simulations. We support the analytical results with several numerical experiments in DUNE-FEM. The objective in Chapter 4 is to investigate the rate of convergence by looking at the L^2 and H^1 error. For the L^2 error, one would predict a linear rate in the deformation due to the pressure term depending on the unit normal. We use two examples using two different initial surfaces to test the convergence rates.

In all cases, the rate in deformation was better than expected, namely quadratic. The theory suggested only linear rate which means these results are likely example specific.

The first example (Section 4.1) is a unit sphere. Here the approximation of the unit normal will be very good, so it seems reasonable to obtain better convergence rates. In the case of Dziuk surface in Section 4.2 however, we do not observe a reduced rate of the deformation either. This is surprising due to the surface not even being convex. The H^1 errors still converged at rate one as expected. We would ideally carry out more examples to find the source of our convergence rates. One option is to reduce the problem's range to 1 (or 2 in including $\mathbf{w}$) to reduce the size of the matrix and hence the computational cost of inverting the system.

Later in the section, we focus on the application of the model in a more realistic setting, studying, in particular, the effect of realistic model parameters. We consider two examples. The first is a discocyte shape and the second a pear shaped surface. In both cases, a similar effect is seen under increase/decreases in the parameters. The outcomes are driven by the relative size of the linker and pressure with the tension being an influence of the bleb size and location.

Beyond the parameter study, we discuss the implementation of the model and the existing literature. We compare the model in this thesis to the one for curves in [7]. We find that they do in fact completely agree to produce a tiny relative error. This shows our model correctly extends the underlying model found in [38] to 3D.

Finally, we demonstrated the ability to simulate three-dimensional cell shapes. We postulate this could be very useful to a biologist who has access to complex imaging data, in particular, the DUNE-femPy implementation of the model (Section 4.5.1). This is the same version of the model seen previously but now coded via Python which is then fed into DUNE-Fem. Advantages of the approach include a much easier learning curve and the starting position for users without advanced numerical backgrounds. At a basic level, the user only needs to edit the parameter value and grid to begin running own examples. If the user has knowledge of bilinear forms the script is much easier to adapt than most other numerical environments. If the person wishes to trial new non-linear formulations this is also easier than in DUNE-fem directly.

Our approach could be further extended to include more complex functions that incorporate more sophisticated behaviour. For example the reconstruction of the cortex and potential attachment of linkers. It would be best in such a formulation to move from the reference surface Γ^0 and instead use the moving surface $\Gamma(t)$, an evolving domain. Taking such a step would make the analysis more challenging due

to the metric now being time dependent. Simulation maybe achievable as mean curvature flow is already implemented in Dune-fempy. Similar force terms such as bending and tension could be made for such a model to mimic similar physical laws used here, one such example is [18].

The effect of these terms could be greatly explored. The breaking of linkers happens instantaneously, bounds may weaken before tearing as the membrane-cortex gap gets larger. This would mostly affect the growth of blebs as the largest effect will be felt on the boundary of a protrusion.

Another option is to model the pressure differently. We restricted our study to the case of using only the normal at $t = 0$. This simplification allowed us to obtain analytical results, since the problem could not become degenerate. Another possibility is the size of the pressure term. One could use the evolving volume instead of assuming it to be constant. We do not expect that this would change simulation results significantly since all expansion would decrease the pressure. A change would be required if the model was to be extended to replicate more of the bleb cycle such as the expansion of blebs. In our simulation so bubble like protrusion ever formed and the dominate term once the membrane is detached is the pressure.

In this thesis, we took an existing 2D finite difference model and re-derived it for 3D in a variational framework. The model was then generalised and analysed by showing well posedness and some a priori estimates. We also formulated our own fully discrete scheme and showed stability bounds. After this, we implemented the problem in Dune and showed convergence rates of the solution in two examples. This implementation was then used to create some more realistic examples by showing how the model predicts bleb sites for two initial membrane shapes. Finally, we are able to show some consistency between our model and the original 2D finite difference model.

Bibliography

[1] R. Alert and J. Casademunt. Bleb nucleation through membrane peeling. *Physical Review Letters*, 116:068101, 2016.

[2] A. Alphonse, C.M. Elliott, and B. Stinner. An abstract framework for parabolic PDEs on evolving spaces. *Portugaliae Mathematica*, 72:1–46, 2015.

[3] U. Ayachit. The paraview guide: A parallel visualization application. *Kitware Inc, USA*, 2015.

[4] Maugis B., J. Brugués, P. Nassoy, N. Guillen, P. Sens, and F. Amblard. Dynamic instability of the intracellular pressure drives bleb-based motility. *Journal of Cell Science*, 123:3884–3892, 2010.

[5] P. Baniukiewicz, S. Collier, and T. Bretschneider. QuimP: analyzing transmembrane signalling in highly deformable cells. *Bioinformatics*, 34(15):2695–2697, 2018.

[6] R. Barreira, C.M. Elliott, and A Madzvamuse. The surface finite element method for pattern formation on evolving biological surfaces. *Journal of Mathematical Biology*, 63:1095–1119, 2011.

[7] T. Bretschneider, S. Collier, P. Paschke, and R.R. Kay. Image based modeling of bleb site selection. *Scientific Reports*, 7:66–92, 2017.

[8] G. Charras and E. Paluch. Blebs lead the way: how to migrate without lamellipodia. *Nature Reviews Molecular Cell Biology*, 9:730–736, 2008.

[9] Y. Chen and C.B. Macdonald. The Closest Point Method and multigrid solvers for elliptic equations on surfaces. *SIAM Journal on Scientific Computing*, 37(1), 2015.

[10] K. Deckelnick, G. Dziuk, and C.M. Elliott. Computation of geometric partial differential equations and mean curvature flow. *Acta Numerica*, 14, 2005.

[11] A. Dedner, P. Madhavan, and B. Stinner. Analysis of the discontinuous Galerkin method for elliptic problems on surfaces. *IMA Journal of Numerical Analysis*, 33:952–972, 2013.

[12] A. Dedner, M. Nolte, M. Ohlberger, and R. Klöfkorn. A generic interface for parallel and adaptive scientific computing: Abstraction principles and the DUNE-FEM module. *Computing*, 90:165–196, 2011.

[13] A. Diz-Muoz, M. Krieg, M. Bergert, I. Ibarlucea-Benitez, D.J. Muller, E. Paluch, and C. Heisenberg. Control of directed cell migration in vivo by membrane-to-cortex attachment. *PLoS Biology*, 8:1–12, 2010.

[14] G. Dziuk. Finite elements for the Beltrami operator on arbitrary surfaces. *Partial Differential Equations and Calculus of Variations*, pages 142–155, 1988.

[15] G. Dziuk and C.M. Elliott. Finite elements on evolving surfaces. *IMA Journal of Numerical Analysis*, 25:385–407, 2007.

[16] G. Dziuk and C.M. Elliott. Surface finite elements for parabolic equations. *Journal of Computational Mathematics*, 25:385–407, 2007.

[17] G. Dziuk and C.M. Elliott. Finite element methods for surface PDEs. *Acta Numerica*, 22:289–396, 2013.

[18] C M. Elliott and B. Stinner. Modeling and computation of two phase geometric biomembranes using surface finite elements. *Journal of Computational Physics*, 229:6585–6612, 2010.

[19] C.M. Elliott and T. Ranner. Evolving surface finite element method for the Cahn-Hilliard equation. *Numerische Mathematik*, 129:483–534, 2015.

[20] E. Evans and E. Sackmann. Translational and rotational drag coefficients for a disk moving in a liquid membrane associated with a rigid substrate. *Journal of Fluid Mechanics*, 194:553–561, 1988.

[21] P. Friedl. Prespecification and plasticity: shifting mechanisms of cell migration. *Current Opinion in Cell Biology*, 1:14–23, 2004.

[22] D. Gilbarg and N.S. Trudinger. *Elliptic Partial Differential Equations of Second Order*. Springer, 1977.

[23] M. Hallett and S. Dewitt. Ironing out the wrinkles of neutrophil phagocytosis. *Trends in Cell Biology*, 17:209–214, 2007.

[24] W. Helfrich. Elastic properties of lipid bilayers: theory and possible experiments. *Zeitschrift für Naturforschung. Teil C: Biochemie, Biophysik, Biologie, Virologie*, 28, 1973.

[25] M. Kass, A. Witkin and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1988.

[26] E. Paluch and E. Raz. Role of cortical tension in bleb growth. *National Academy of Sciences*, 106:18581–18586, 2009.

[27] E. Paluch and E. Raz. The role and regulation of blebs in cell migration. *Current Opinion in Cell Biology*, 25:582–590, 2013.

[28] G. A. Ponuwei. Unmasking plasma membrane blebbing. *Journal of Biomedical Science and Applications*, 1, 2017.

[29] J.C. Robinson. Infinite-dimensional dynamical systems: An introduction to dissipative parabolic pdes and the theory of global attractors. *Cambridge Press*, 2001.

[30] V. Sanz-Moreno and C. Marshall. The plasticity of cytoskeletal dynamics underlying neoplastic cell migration. *Current Opinion in Cell Biology*, 22:690–696, 2010.

[31] S.L. Schuster, F.J. Segerer, F.A. Gegenfurtner, K. Kick, C. Schreiber, M. Albert, A.M Vollmar, J.O. Rädler, and S. Zahler. Contractility as a global regulator of cellular morphology, velocity and directionality in low-adhesive fibrillary micro-environments. *Biomaterials*, 102, 2016.

[32] J. Simon. Compact sets in the space $L_p(O,T;B)$. *Annali di Matematica Pura ed Applicata*, 146:65–96, 1986.

[33] B. Stinner, C.M. Elliott, and C. Venkataraman. Modelling cell motility and chemotaxis with evolving surface finite elements. *Journal of the Royal Society*, 9:3027–3044, 2012.

[34] W. Strychalski and R. D. Guy. A computational model of bleb formation. *Mathematical Medicine and Biology*, 30 2:115–130, 2013.

[35] A. Taloni, E. Kardash, O. U. Salman, L. Truskinovsky, S. Zapperi, and C.A.M. La Porta. Volume changes during active shape fluctuations in cells. *Physical Review Letters*, 114:208101, 2015.

[36] J. Tinevez, U. Schulze, G. Salbreux, J. Roensch, and J. Joanny. Role of cortical tension in bleb growth. *National Academy of Sciences*, 106:18581–18586, 2009.

[37] L. N. Trefethen and D. Bau III. *Numerical Linear Algebra*. SIAM, 1997.

[38] R.A. Tyson, E. Zatulovskiy, R.R. Kay, and T. Bretschneidera. How blebs and pseudopods cooperate during chemotaxis. *Proceedings of the National Academy of Sciences*, 111:11703–11708, 2014.

[39] Hans Wilhelm Alt. *Linear Functional Analysis: An application-Oriented Introduction*. Springer, 2016.

[40] Inc. Wolfram Research. Mathematica. 2018.

[41] T. Wooley, E. Gaffney, J. Oliver, R. Baker, and A. Goriely. Three mechanical models for blebbing and multi-blebbing. *IMA Journal of Numerical Analysis*, 79:636–660, 2014.

[42] T. Wooley, E. Gaffney, J. Oliver, R. Baker, S. Waters, and A. Goriely. Cellular blebs: pressure-driven, axisymmetric, membrane protrusions. *Biomechanics and Modeling in Mechanobiology*, 13:463–476, 2013.

[43] Kunito Yoshida and Thierry Soldati. Dissection of amoeboid movement into two mechanically distinct modes. *Journal of Cell Science*, 119:3833–3844, 2006.

[44] J. Young and S. Mitran. A numerical model of cellular blebbing: A volume-conserving, fluid-structure interaction model of the entire cell. *Journal of biomechanics*, 43:210–230, 2009.

[45] E. Zatulovskiy, R. Tyson, T. Bretschneider, and R. Kay. Bleb-driven chemotaxis of dictyostelium cells. *Journal of Cell Biology*, 204:1027–1044, 2014.

Appendix A

Additional Theorems and Definitions

Definition A.0.1. *Let $f(s)$ a differentiable function defined over a discrete interval D divided into N points labelled s_i, all distance Δs apart. We define the finite difference approximation of the spacial derivative as follows*

$$\partial_{s^2} f(s_i) := \frac{f(s_{i+1}) - 2f(s_i) + f(s_{i-1})}{\Delta s^2} \tag{A.1}$$

$$\partial_{s^4} f(s_i) := \frac{f_{s,s}(s_{i+1}) - 2f_{s,s}(s_i) + f_{s,s}(s_{i-1})}{\Delta s^2} \tag{A.2}$$

Lemma A.0.1. *(Lebesgue's convergence theorem) let $f_k, f : S \to Y$ be μ-measurable, let $g_k \to g$ in $L^1(\mu; \mathbb{R})$ as $k \to \infty$ and let $1 \leq p \leq \infty$. Suppose that*

$$\begin{aligned} f_k &\to f && \mu\text{-almost everywhere as } k \to \infty \\ |f_k|^p &\leq g_k && \mu\text{-almost everywhere for all } k \in \mathbb{N} \end{aligned}$$

Then it follows that $f_k, f \in L^p(\mu; Y)$ and $f_k \to f$ in $L^p(\mu; Y)$ as $k \to \infty$

Proof. See 3.25 on page 60 in [39] □

Lemma A.0.2. *(Gronwall's Inequality) Let $z(\cdot)$ be a non negative, absolutely continuous function on $[0, T]$, which satisfies for a.e t the differential inequality*

$$z'(t) \leq \phi(t)z(t) + \psi(t),$$

where $\phi(t)$ and $\psi(t)$ are non negative, summable functions on $[0, T]$. Then

$$z(t) \leq e^{\int_0^t \phi(s)ds} \left[z(0) + \int_0^t \psi(s)ds \right] \qquad \forall 0 \leq t \leq T$$

Lemma A.0.3. *(Discrete Gronwall's Inequality) let $z_k \geq 0$, $k = 0, 1, 2, .., n, .., N$ satisfy*

$$z_{n+1} \leq \lambda z_n + \lambda G$$

where $\lambda > 0$ and $G > 0$. then

$$z_n \leq \lambda^n z_0 + \frac{1 - \lambda^n}{1 - \lambda} \lambda G, \qquad \lambda \neq 1$$
$$z_n \leq z_0 + nG, \qquad \lambda = 1$$